

Εισαγωγή στο ΕΛΕΥΘΕΡΟ ΛΟΓΙΣΜΙΚΟ

ΣΥΓΓΡΑΦΕΙΣ:

J. Gonzalez Barahona
J. Seone Pascual
G. Robles

ΣΥΝΤΟΝΙΣΤΕΣ:

J. Mas Hernandez
D. Megias Jimenez

**Jordi Mas Hernández**

Ιδρυτικό μέλος της Softcatalà και του Τηλεματικού δικτύου RedBBS. Εργάστηκε ως σύμβουλος (*Consultant*) σε εταιρείες όπως η Menta, η Telépolis, η Vodafone, η Lotus, η eresMas, η Amena και η Terra España.

**David Megías Jiménez**

Μηχανικός Ηλεκτρονικών Υπολογιστών (*Computer Science Engineer*) στο Ανοικτό Πανεπιστήμιο της Βαρκελώνης (*Universitat Autònoma de Barcelona, UAB*, Ισπανία). Κάτοχος Master σε θέματα Advanced Process Automation Techniques του UAB. Κάτοχος PhD στις Επιστήμες υπολογιστών (*Computer Science*), του UAB. Αναπληρ. Καθηγητής στο Τμήμα Επιστήμης Υπολογιστών (*Computer Science*), Πολυμέσων και Τηλεπικοινωνιών, του Ανοικτού Πανεπιστημίου της Καταλωνίας (*Universitat Oberta de Catalunya, UOC*, Ισπανία) και Διευθυντής του Προγράμματος Master στο Ελεύθερο Λογισμικό του UOC.

**Jesús M. González-Barahona**

Αναπληρ. Καθηγητής στο Τμήμα Τηλεματικών Συστημάτων και Computing, του Πανεπιστημίου του Βασιλέως Juan Carlos (Μαδρίτη, Ισπανία), όπου συντονίζει την ομάδα έρευνας GsyC/LibreSoft. Εστιάζει τα επαγγελματικά του ενδιαφέροντα στη μελέτη της ανάπτυξης του Ελεύθερου Λογισμικού και στη μεταφορά της γνώσης από αυτό το γνωστικό πεδίο στον βιομηχανικό τομέα.

**Joaquín Seoane Pascual**

Αναπληρ. Καθηγητής στο Τμήμα Μηχανικών Τηλεματικών Συστημάτων του Τεχνικού Πανεπιστημίου Μαδρίτης (Ισπανία), όπου δίδαξε μαθήματα προγραμματισμού, πρωτοκόλλων, αρχιτεκτονικής υπολογιστών, λειτουργικών συστημάτων, υπηρεσιών Internet, βάσεων δεδομένων, διαχείρισης συστημάτων και Ελεύθερου Λογισμικού. Τα τρέχοντα ενδιαφέροντά του περιλαμβάνουν την εφαρμογή της Τεχνολογίας των Υπολογιστών και Πληροφορικής (ICT) σε απομονωμένες περιοχές αναπτυσσόμενων χωρών.

**Gregorio Robles**

Αναπληρ. Καθηγητής στο Πανεπιστήμιο του Βασιλέως Juan Carlos (Madrid, Ισπανία), όπου εκπόνησε και το θέμα του PhD τον Φεβρουάριο του 2006. Πέρα από τα διδακτικά του καθήκοντα, ασχολείται με έρευνα σχετικά με την ανάπτυξη Ελεύθερου Λογισμικού από την σκοπιά του software engineering, με ιδιαίτερη εστίαση σε θέματα ποσοτικοποίησης (*quantitative issues*).

Τρίτη έκδοση: Σεπτέμβριος 2009

Fundació per a la Universitat Oberta de Catalunya.

Av. Tibidabo, 39-43, 08035 Βαρκελώνη

Το υλικό προετοιμάστηκε από την Eureka Media, SL

© Jesús M. González Barahona, Joaquín Seoane Pascual, Gregorio Robles

Copyright © 2010, FUOC. Δίδεται άδεια για αντιγραφή, διανομή και τροποποίηση αυτού του κειμένου, είτε υπό τους όρους της Αδείας GNU Free Documentation Licence, Version 1.2 είτε οποιασδήποτε επόμενης έκδοσης, δημοσιευμένης από το Ίδρυμα Ελεύθερου Λογισμικού (Free Software Foundation), χωρίς “invariant sections or front-cover or back-cover texts”, είτε υπό τους όρους της αδείας Creative Commons by-sa 3.0 license, κατ' επιλογή του χρήστη. Ένα αντίτυπο αυτών των αδειών περιλαμβάνεται στα αντίστοιχα παραρτήματα αυτού του κειμένου.

Πρόλογος

Το λογισμικό κατέστη, κατά τις τελευταίες λίγες δεκαετίες, ένας στρατηγικός κοινωνικός πόρος. Η ανάδυση του Ελεύθερου Λογισμικού, το οποίο διείσδυσε σε καίριους τομείς της αγοράς της Πληροφορικής και των Υπολογιστών (ICT), προκαλεί μία αλλαγή των οικονομικών παραμέτρων που αφορούν την ανάπτυξη και χρήση του λογισμικού. Το Ελεύθερο Λογισμικό- που, μερικές φορές, αναφέρεται και ως “Ανοικτού Κώδικα” ή ως “Libre Software”- μπορεί να χρησιμοποιηθεί, να μελετηθεί, να αντιγραφεί, να τροποποιηθεί και να διανεμηθεί ελεύθερα. Προσφέρει την ελευθερία να μπορούμε να μάθουμε και να διδάξουμε, χωρίς να εγκλωβιζόμαστε σε εξαρτήσεις από κανέναν συγκεκριμένο προμηθευτή τεχνολογίας. Αυτές οι ελευθερίες θεωρούνται ως βασική προϋπόθεση για την επίτευξη μίας διατηρήσιμης ανάπτυξης και μίας κοινωνίας “ολικής πληροφορίας” (*inclusive information society*).

Παρ' όλο το αυξανόμενο ενδιαφέρον για τις ελεύθερες τεχνολογίες (Ελεύθερο Λογισμικό και Ανοιχτά Πρότυπα), ακόμη και τώρα, είναι περιορισμένος ο αριθμός ανθρώπων με επαρκή γνώση και εξειδίκευση σε αυτούς τους τομείς. Η Ακαδημία Ελεύθερης Τεχνολογίας (*Free Technology Academy, FTA*) προσπαθεί να καλύψει αυτή τη ζήτηση.

Εισαγωγή στην FTA

Η Ακαδημία Ελεύθερης Τεχνολογίας (*Free Technology Academy, FTA*) συνιστά μία κοινή πρωτοβουλία, διάφορων εκπαιδευτικών Ινστιτούτων, από διάφορες χώρες. Αποσκοπεί να συνεισφέρει στη δημιουργία μίας κοινωνίας που να επιτρέπει σε όλα τα μέλη της να μελετούν, να συμμετέχουν και να κτίζουν πάνω στην προϋπάρχουσα γνώση, χωρίς αποκλεισμούς.

Τι προσφέρει η FTA;

Η Ακαδημία προσφέρει ένα online πρόγραμμα μαθημάτων, σε επίπεδο master, διαρθρωμένο σε διάφορες ενότητες μαθημάτων για τις Ελεύθερες Τεχνολογίες. Οι εκπαιδευόμενοι μπορούν να επιλέξουν τη συμμετοχή τους, είτε σε ατομικό πρόγραμμα μαθημάτων, είτε στο ολοκληρωμένο πρόγραμμα. Η επίβλεψη (*tuition*) γίνεται online, στο εικονικό FTA campus και υποστηρίζεται από Διδακτικό προσωπικό προερχόμενο από τα συνεργαζόμενα Πανεπιστήμια. Τα credits που παίρνει ο εκπαιδευόμενος, στα πλαίσια του προγράμματος FTA, αναγνωρίζονται από όλα αυτά τα Πανεπιστήμια.

Ποιος βρίσκεται πίσω από την FTA;

Η FTA άρχισε το 2008, υποστηριζόμενη από το Πρόγραμμα της Διά βίου Εκπαίδευσης (*Life Long Learning Programme, LLP*) της Ευρωπαϊκής Επιτροπής, με τον συντονισμό του Ινστιτούτου Ελεύθερης Γνώσης (*Free Knowledge Institute*), σε συνεργασία με 3 Ευρωπαϊκά Πανεπιστήμια: το Ανοικτό Πανεπιστήμιο Ολλανδίας, το Ανοικτό Πανεπιστήμιο Καταλωνίας (Ισπανία) και το Πανεπιστήμιο του Agder (Νορβηγία).

Σε ποιον απευθύνεται η FTA;

Η Free Technology Academy απευθύνεται ιδιαίτερα προς τους επαγγελματίες της Πληροφορικής (IT), τους Δασκάλους και Καθηγητές, τους μαθητές, τους φοιτητές και προς τους έχοντες την ευθύνη λήψης των αποφάσεων (*decision makers*).

Τι γίνεται σχετικά με τις άδειες;

Όλα το διδακτικό υλικό που χρησιμοποιείται από- και ανεπτύχθη από- την FTA, αποτελείται από Ανοικτούς Εκπαιδευτικούς Πόρους (*Open Educational Resources*) που δημοσιεύθηκαν με ελεύθερες άδειες “copyleft”, οι οποίες επιτρέπουν ελεύθερη χρήση, τροποποίηση και αναδιανομή. Παρομοίως, το λογισμικό που χρησιμοποιείται στο εικονικό FTA campus, είναι Ελεύθερο Λογισμικό και βασίζεται πάνω στο framework των Ανοικτών Προτύπων.

Εξέλιξη αυτού του βιβλίου

Η FTA χρησιμοποίησε προϋπάρχον διδακτικό υλικό, από μαθήματα του ανοικτού Πανεπιστημίου της Καταλωνίας, το οποίο είχε αναπτυχθεί σε συνεργασία με το προσωπικό της ομάδας LibreSoft, από το Πανεπιστήμιο Rey Juan Carlos. Το 2008, αυτό το βιβλίο μεταφράστηκε στα Αγγλικά, με τη βοήθεια του SELF Project (*Science, Education & Learning in Freedom*), που υποστηριζόταν από το 6^ο Πλαίσιο Στήριξης της Ευρωπαϊκής Επιτροπής. Το 2009, αυτό το υλικό βελτιώθηκε από την FTA. Η FTA ανέπτυξε , επιπλέον, έναν οδηγό σπουδών και εκπαιδευτικές δραστηριότητες, που είναι στη διάθεση των εκπαιδευομένων που εγγράφονται στο FTA Campus.

Συμμετοχή

Αυτοί που χρησιμοποιούν τα εκπαιδευτικά υλικά της FTA ενθαρρύνονται να εκφράζουν τη γνώμη τους και να κάνουν υποδείξεις για τη βελτίωση του προγράμματος. Στην ιστοσελίδα της FTA έχει προβλεφθεί ένας ειδικός χώρος γι' αυτό το feedback. Αυτές οι υποδείξεις θα ληφθούν υπ' όψη στις επόμενες εκδόσεις. Επιπλέον, η FTA παροτρύνει τους πάντες να χρησιμοποιήσουν και να

διαδώσουν περαιτέρω αυτό το υλικό, καθώς και να φτιάξουν νέες εκδόσεις και μεταφράσεις.

Για περισσότερες ενημερωμένες και λεπτομερείς πληροφορίες για το βιβλίο αυτό, περιλαμβανομένων των μεταφράσεων και άλλων formats, συνδεθείτε στη διεύθυνση:

<http://ftacademy.org/materials/fsm/1>. Για περαιτέρω πληροφορίες σχετικά με την εγγραφή στο πρόγραμμα μαθημάτων online της FTA, σας παρακαλούμε να επισκεφθείτε τον ιστότοπο της Ακαδημίας: <http://ftacademy.org/>

Ελπίζω ειλικρινά αυτό το διδακτικό βιβλίο να σας βοηθήσει στην προσωπική σας διαδρομή εκπαίδευσης/ μάθησης και για να βοηθήσετε και άλλους, στη δική τους αναζήτηση. Προσδοκώ να σας δω να συμμετέχετε σε ένα από τα διάφορα κινήματα ελεύθερης γνώσης και ελεύθερης τεχνολογίας!

Happy learning!

Wouter Tebbens
Πρόεδρος του Free Knowledge Institute
Διευθυντής της Free Technology Academy

Οι συγγραφείς θα ήθελαν να ευχαριστήσουν το Ίδρυμα του Ανοικτού Πανεπιστημίου Καταλωνίας, για τη χρηματοδότηση, τόσο της 1^{ης} έκδοσης αυτού του έργου, όσο και ενός μεγάλου μέρους των βελτιώσεων που οδήγησαν στη 2^η έκδοση, ως μέρος του Προγράμματος Master στο Ελεύθερο Λογισμικό, που προσφέρει το εν λόγω Πανεπιστήμιο, όπου και χρησιμοποιείται ως υλικό, για διάφορα μαθήματα.

Η μετάφραση αυτού του έργου στα Αγγλικά κατέστη εφικτή με την υποστήριξη εκ μέρους του SELF Project, της SELF Platform, του προγράμματος της Ευρωπαϊκής Επιτροπής για τις Τεχνολογίες της Κοινωνίας της Πληροφορίας και του Ανοικτού Πανεπιστημίου Καταλανίας. Θα θέλαμε να ευχαριστήσουμε τη lexia:park για τη μετάφραση των υλικών που έκανε στα Αγγλικά.

Η τρέχουσα έκδοση του υλικού αυτού, στα Αγγλικά, έχει επεκταθεί με τη χρηματοδότηση από το *Free Technology Academy (FTA) project*. Το FTA project έλαβε στήριξη από την Ευρωπαϊκή Επιτροπή (αριθμός αναφοράς 142706- LLP-1-2008-1-NL-ERASMUS-EVC του Προγράμματος για Διά βίου Μάθηση). Αυτό το δημοσιευμένο έργο εκφράζει αποκλειστικά τις απόψεις των συγγραφέων και η Ευρωπαϊκή Επιτροπή δε μπορεί να θεωρηθεί υπεύθυνη για οποιαδήποτε χρήση που μπορεί να γίνει, της πληροφορίας που περιέχεται σε αυτό.

Πρόλογος Μεταφραστή:

Ο ενθουσιασμός μου για το Ελεύθερο Λογισμικό με έφερε κάποια στιγμή στην “ανακάλυψη” της Free Technology Academy και του παρόντος βιβλίου, που θεωρώ ότι συμπυκνώνει με εξαιρετικό τρόπο την ιστορία, τη φύση, τις ιδιαιτερότητες και τα οφέλη που μπορεί να προσφέρει το Ελεύθερο Λογισμικό, τόσο ατομικά στον καθένα μας, όσο και, ειδικότερα, στο Δημόσιο Τομέα.

Μετά από κάποια αναζήτηση, κατέληξα στο συμπέρασμα πως δεν υπήρχε κανένα άλλο ανάλογο βιβλίο στην Ελληνική βιβλιογραφία, που να εξηγεί εύγλωττα και επιστημονικά το θέμα αυτό. Ξεκίνησα, λοιπόν, το “τρελό” εγχείρημα της πλήρους μετάφρασης του κειμένου στα Ελληνικά, παρά και την επίγνωση των ανεπαρκών προσωπικών μου γνώσεων στην Πληροφορική. Έτσι, δεν απετέλεσε έκπληξη το ότι συνάντησα σημαντικές δυσκολίες στη μεταφραστική μου προσπάθεια, ούτε το γεγονός ότι ο “μαραθώνιος” αυτός διήρκεσε σχεδόν 10 μήνες, συμπίεσμένος στον ελάχιστο προσωπικό ελεύθερο χρόνο.

Για τους παραπάνω λόγους, θα παρακαλούσα τους αναγνώστες να δείξουν κάποια επιείκεια για τις μεταφραστικές ανακρίβειες του Ελληνικού κειμένου. Ακόμη καλύτερα, θα εκτιμούσα απεριόριστα τη συνδρομή όλων, με παρατηρήσεις, διορθώσεις ή προτάσεις, που μπορείτε να απευθύνετε στην ηλεκτρονική διεύθυνση: kvisitor@gnu.gr.

Ως συνέπεια των δυσκολιών που συνάντησα, έγιναν οι εξής επιλογές:

- 1) στις ιδιαίτερα δύσκολες λέξεις, άφησα σε παρένθεση τον αρχικό Αγγλικό όρο, για να μπορεί ο αναγνώστης να συμπεράνει και μόνος του την καλύτερη δυνατή αντιστοιχία.
- 2) Άφησα αμετάφραστο τόσο το κείμενο της Άδειας GPL, όσο και εκείνο της Άδειας Creative Commons, καθώς περιέχουν ιδιαίτερα λεπτές νομικές έννοιες και, προφανώς, για ένα τέτοιο εγχείρημα απαιτούνται οι αντίστοιχες γνώσεις. Τα κείμενα όμως των Αδειών, αποτελούν Παραρτήματα του βιβλίου, οπότε η παράλειψη αυτή δε νομίζω πως περικόπτει τίποτε από την αξία του βιβλίου ή από την ευχαρίστηση της ανάγνωσής του.
- 3) Καθώς ο στόχος είναι να μπορούν όλοι να έχουν πρόσβαση στο κείμενο αυτό, αποφάσισα να το αποθηκεύσω σε μορφή αρχείου .PDF, για να μπορεί άνετα να το διαβάσει ο καθένας, όποιο Λειτουργικό Σύστημα και αν χρησιμοποιεί.

Ελπίζω το καταπληκτικό αυτό κείμενο να αποτελέσει σημείο αναφοράς για όσους επιθυμούν να γνωρίσουν συνολικά τον κόσμο του Ελεύθερου Λογισμικού, να βοηθήσει τους Εκπαιδευτικούς όλων των βαθμίδων να μεταβιβάσουν και στη νεολαία τις γνώσεις αυτές, καθώς και να συνδράμει το πολιτικό προσωπικό με θέσεις ευθύνης στη Δημόσια Διοίκηση να υιοθετήσουν λύσεις Ελεύθερου Λογισμικού στο Δημόσιο βίο.

Ευχαριστίες:

Θα ήθελα να εκφράσω τις ευχαριστίες μου στο Σύλλογο Ελλήνων Φίλων και χρηστών Ελεύθερου Λογισμικού (GreekLUG, www.greeklug.gr), διότι στην προσπάθειά μου να κατανοήσω το Κίνημα του Ελεύθερου Λογισμικού, μου προσέφεραν όχι μόνον άφθονες συμβουλές και πολύτιμα ερεθίσματα, αλλά φιλοξενούν στο server του Συλλόγου την παρούσα έκδοση, από όπου μπορεί ο καθένας να το κατεβάσει και να το απολαύσει:

http://olympus.greeklug.gr/uploads/fta/fta-intro_floss_el.pdf

Ιδιαίτερες ευχαριστίες οφείλω, επίσης, προς τον αγαπητό κ°. Ιωάννη Σταμέλο, Αναπληρωτή Καθηγητή του Τμήματος Πληροφορικής του Α.Π.Θ., ο οποίος δέχθηκε ευγενέστατα να προλογίσει την Ελληνική έκδοση του βιβλίου, αλλά και να προσφέρει τις συμβουλές του και υποδείξεις του για τη μετάφραση αυτή.

Ευχαριστίες, τέλος, εκφράζω και προς την αγαπητή κ°. Έλενα Σπυροπούλου, διακεκριμένη Νομικό, ευρύτερα προβεβλημένη για τις γνώσεις της των νομικών πτυχών του ΕΛ/ΛΑΚ, η οποία επίσης δέχθηκε να προσφέρει τις πολύτιμες υποδείξεις της για τα κεφάλαια εκείνα του βιβλίου που εξετάζουν τα νομικά θέματα.

Άδεια κυκλοφορίας:

Σύμφωνα με τις πρόνοιες της αρχικής έκδοσης, το βιβλίο κυκλοφορεί και διανέμεται υπό διπλή Άδεια, δηλαδή υπό την “GNU Free Documentation Licence, Έκδοση 1.2 ή και οποιωνδήποτε μεταγενέστερων εκδόσεωνείτε υπό τους όρους της Αδειάς Creative Commons by-sa 3.0, κατ' επιλογήν του χρήστη”. Με το ίδιο μοντέλο διπλής αδειοδότησης θα συνεισφέρω και εγώ την παρούσα Ελληνική έκδοση.

ΠΡΟΛΟΓΟΣ ΕΛΛΗΝΙΚΗΣ ΕΚΔΟΣΗΣ

Με ιδιαίτερη χαρά προλογίζω την ελληνική έκδοση του βιβλίου αυτού που αποτελεί ένα σταθμό για τη διάδοση και στήριξη του Ελεύθερου Λογισμικού και των Ελεύθερων Τεχνολογιών γενικότερα. Το βιβλίο έρχεται να καλύψει ένα κενό που υπάρχει στην ελληνική βιβλιογραφία, η οποία στο πεδίο των Ελεύθερων Τεχνολογιών είναι πραγματικά ισχνή, τη στιγμή που γράφονται αυτές οι γραμμές.

Το βιβλίο έχει συγγραφεί από ανθρώπους που έχουν πρωτοστατήσει στο ευρωπαϊκό κίνημα για το Ελεύθερο Λογισμικό και έχουν, ήδη, γράψει τη δική τους ιστορία σε αυτό. Οι συγγραφείς είναι Ισπανοί ακαδημαϊκοί, που έχουν ταυτισθεί με το Ελεύθερο Λογισμικό, έχουν συμμετάσχει σε αυτό, το έχουν ερευνήσει, προσφέροντας σημαντικά ερευνητικά αποτελέσματα και, τελικά, του έχουν προσφέρει πάρα πολλά. Ένα δείγμα ακριβώς αυτής της προσφοράς τους στο Ελεύθερο Λογισμικό, στο πλαίσιο της Ακαδημίας Ελεύθερων Τεχνολογιών, έχουμε σήμερα τη δυνατότητα να απολαύσουμε όλοι εμείς που ενδιαφερόμαστε για ποιοτικό και αποτελεσματικό λογισμικό, χωρίς να γινόμαστε πειθαναγκασμένοι χρήστες αμφίβολης αξίας και πανάκριβων εφαρμογών και “εργαλείων αύξησης της παραγωγικότητας”.

Θα πρέπει όλοι μας να ευχαριστήσουμε θερμά τον Κώστα Μουσαφείρη, που αφιέρωσε τον ελεύθερο χρόνο του επί δέκα μήνες από τη ζωή του, για να μπορέσει να μεταφράσει το έργο αυτό από τα Αγγλικά στα Ελληνικά. Μία εθελοντική προσφορά τέτοιας έκτασης, όπως αυτής του αγαπητού Κώστα, μόνο ελπίδα και θάρρος μπορεί να δώσει για τη συνέχιση και παραπέρα διάδοση και επιτυχία αυτού του καταπληκτικού από πολλές απόψεις φαινομένου που ακούει στο όνομα Ελεύθερο Λογισμικό. Ελπίζω το παράδειγμά του να βρει πολλούς μιμητές και να γίνει η αρχή για τη δημιουργία μίας πλούσιας Ελληνικής βιβλιογραφίας στο θέμα αυτό.

Τελειώνοντας, θα ήθελα να συμβουλέψω τον αναγνώστη να θεωρήσει τις γνώσεις και πληροφορίες που θα πάρει από το βιβλίο που έχει στα χέρια ή στον υπολογιστή του, σαν την αρχή μόνο της γνωριμίας του με το Ελεύθερο Λογισμικό και να του ζητήσω/ευχηθώ να εμβαθύνει περισσότερο σε ένα φαινόμενο που, χωρίς υπερβολή, άλλαξε σε σημαντικό βαθμό τον τρόπο που βλέπουμε σήμερα τις τεχνολογίες του λογισμικού και της πληροφορικής γενικότερα.

Ιωάννης Σταμέλος
Αναπληρωτής Καθηγητής
Τμήμα Πληροφορικής
Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης

Εισαγωγή

"Ο,ποιος το ακούει αυτό, αν μπορεί να τραγουδήσει, τότε δικαιούται να προσθέσει και να το αλλάξει κατά βούληση. Άστο να πάει από χέρι σε χέρι: άσε να το έχουν αυτοί που το ζητούν. Σαν μία μπάλα ανάμεσα σε νέες γυναίκες, πιάστην αν μπορείς. Αφού αυτό είναι της "Καλής Αγάπης", δάνεισέ το με χαρά: μη κοροϊδεύεις το όνομά του κρατώντας το κρυφό. Ούτε να το ανταλλάξεις με χρήματα πουλώντας το ή αγοράζοντάς το. Διότι η "Καλή Αγάπη", όταν αγοράζεται, χάνει όλη τη γοητεία της."

Juan Ruiz, Αρχιμανδρίτης της Hita. *Το Βιβλίο της Καλής Αγάπης* (14^{ος} αιώνας, αρχικό κείμενο στα Αρχαία Ισπανικά)

Η πρώτη έκδοση αυτών των σημειώσεων γράφτηκε από τους Jesús M. González-Barahona, Joaquín Seoane Pascual και Gregorio Robles, ανάμεσα στον Απρίλη και τον Σεπτέμβρη του 2003. Αν και το συζητούσαμε επί αρκετό καιρό να γράψουμε ένα τέτοιο κείμενο, με μαθήματα περί Ελεύθερου Λογισμικού, που ο Joaquín και ο Jesús διδάσκουν στα πλαίσια του προγράμματος PhD των αντίστοιχων τμημάτων τους, το στοιχείο που μας ενθάρρυνε να το αρχίσουμε ήταν η πρωτοβουλία του Ανοικτού Πανεπιστημίου της Καταλωνίας (UOC), που μας ανέθεσε να ετοιμάσουμε υλικό για εισαγωγή σε μαθήματα επιπέδου master, για το Ελεύθερο Λογισμικό. Η εμπλοκή του Jordi Mas (Ακαδημαϊκού συντονιστή των μαθημάτων επιπέδου Master) σε αυτή την αποστολή, υπήρξε καθοριστική, καθώς ήταν εκείνος που μας πρότεινε για τη δουλειά αυτή και μας έφερε σε επαφή με το UOC, πέραν από το ότι υποστήριξε τις σχέσεις μας με το UOC, καθ' όλη τη διάρκεια του project. Σύντομα, μετά την παράδοση της πρώτης έκδοσης, οι συγγραφείς άρχισαν να βελτιώνουν το υλικό, ως μέρος μίας συνεχούς διαδικασίας, αν και με διαφορετική ένταση δραστηριότητας, μέχρι την ολοκλήρωση της δεύτερης έκδοσης, τον Μάιο του 2007. Σε όλο αυτό το διάστημα, η πρώτη έκδοση χρησιμοποιήθηκε εκτενώς στα μαθήματα master για το Ελεύθερο Λογισμικό του UOC, καθώς και σε διάφορα άλλα μεταπτυχιακά μαθήματα, στην Ισπανία και την Αμερική. Η εμπειρία με το UOC κατεγράφη με ιδιαίτερο ενδιαφέρον από τον Gregorio Robles, ο οποίος συμμετείχε σε αυτή και, άρα, απέκτησε σημαντικό feedback, που απεδείχθη ιδιαίτερα πολύτιμο για τη βελτίωση του περιεχομένου. Οι τρεις μας (ο Joaquín, ο Jesús, και από το 2006 και ο Gregorio) συνεχίσαμε και με τη σειρά μεταπτυχιακών μαθημάτων περί λογισμικού στο UPM (*Polytechnic University of Madrid*) και στο URJC (*Rey Juan Carlos University*), εκμεταλλευόμενοι την ευκαιρία να δοκιμάσουμε το υλικό αυτό.

Για μια φορά ακόμη, το UOC ήταν ο καταλύτης γι' αυτή τη δεύτερη έκδοση, αναθέτοντάς μας μία παραγγελία που μας πήρε τόσο πολύ να τελειώσουμε. Η δουλειά των Jordi Mas και David Megías (του UOC), ήταν ουσιαστική και μας παρείχε κρίσιμη υποστήριξη για την προώθηση της νέας έκδοσης. Η δουλειά του José Ignacio Fernández Villamor και του Boni García Gutiérrez, μαθητών του Joaquín Seoane, που συνεργάστηκαν για την αναθεώρηση των υλικών γι' αυτή τη δεύτερη έκδοση, υπήρξε επίσης βασικής σημασίας.

Προηγούμενο υλικό

Μερικά κείμενα σε αυτές τις σημειώσεις, βασίζονται σε προϋπάρχον υλικό, που συνήθως ανήκει στους ίδιους τους συγγραφείς και, ενίοτε, σε τρίτους (χρησιμοποιήθηκαν με την άδειά τους, όταν δεν ξαναγράφτηκαν εντελώς). Ανάμεσά τους, θα θέλαμε να αναφέρουμε τα ακόλουθα (με κίνδυνο να παραλείψουμε και μερικά σημαντικά):

- Υπάρχουν μερικά εδάφια (ειδικά στα κεφάλαια περί Ιστορίας και Οικονομίας) που είναι εμπνευσμένα από το κείμενο *"Free Software/ Open Source: Information Society Opportunities for Europe?"* [132], το οποίο ο Jesús González-Barahona συνδιαμόρφωσε για την Ευρωπαϊκή Επιτροπή. Εν πάσει περιπτώσει, τα εν λόγω εδάφια ανεπτύχθησαν, “πειράχθηκαν” και επικαιροποιήθηκαν, σε τέτοιο βαθμό που, σε πολλές περιπτώσεις, είναι δύσκολα αναγνωρίσιμα.
- Το κεφάλαιο περί μονοπωλίων και Ελεύθερου Λογισμικού (κεφάλαιο 5.4) βασίστηκε στο άρθρο *"Software libre, monopolios y otras yerbas"* ("Ελεύθερο Λογισμικό, μονοπώλια και άλλα χορταρικά") [84], του Jesús M. González-Barahona.
- Τα κεφάλαια για τις νομοθετικές πρωτοβουλίες και τις πρωτοβουλίες της Δημόσιας Διοίκησης για το Ελεύθερο Λογισμικό, βασίστηκαν, εν μέρει, στο *"Iniciativas de las administraciones públicas en relación al Software Libre"* ("Πρωτοβουλίες των Δημοσίων Διοικήσεων σχετικά με το Ελεύθερο Λογισμικό") [103] (ευχαριστούμε τον Pedro de las Heras που μας επέτρεψε να χρησιμοποιήσουμε αυτό το υλικό, το οποίο συν-έγραψε).
- Μέρος του κεφαλαίου για τους λόγους χρήσης του Ελεύθερου Λογισμικού στη Δημόσια Διοίκηση (Κεφάλαιο 6.2), βασίζεται στο άρθρο [85], του Jesús M. González-Barahona.
- Το κεφάλαιο περί Τεχνολογίας Ελεύθερου Λογισμικού, αποτελεί μία προσαρμογή του άρθρου για το state-of-the-art της Τεχνολογίας Λογισμικού, εφηρμοσμένο στο Ελεύθερο Λογισμικό, των Jesús M. González-Barahona και Gregorio Robles, για τα περιοδικά Novática (Ισπανική έκδοση) και Upgrade (Αγγλική έκδοση).
- Στο κεφάλαιο των μελετών περίπτωσης (*case studies*), το κομμάτι για την ανάπτυξη του Linux βασίζεται σε μία παρουσίαση που έκανε ο Juan-Mariano de Goyeneche, σε ένα μεταπτυχιακό

μάθημα "Free Programs", στο Polytechnic University of Madrid, κατά το ακαδημ. έτος 2002-03.

- Το ιστορικό μέρος της λεπτομερούς μελέτης του GNOME, ελήφθη από την ιστορική εισαγωγή που περιλαμβάνεται στο βιβλίο *"Applications development in GNOME2"*, που επιμελήθηκε ο GNOME Hispano και γράφτηκε από έναν από τους συγγραφείς αυτού του βιβλίου.
- Η περιπτωσιολογική μελέτη για το FreeBSD βασίστηκε, εν μέρει, στην παρουσίαση που έκανε ο Jesús Rodríguez στο 3^ο Συνέδριο HispaLinux, που έγινε στη Μαδρίτη, το έτος 2000.
- Οι μελέτες περίπτωσης για το Debian και την Red Hat, βασίστηκαν στην προηγούμενη δουλειά των González-Barahona και συν., που αντιστοίχησαν τα αποτελέσματα της ποσοτικής ανάλυσης αυτών των δύο διανομών με διάφορα άρθρα και δημοσιεύσεις.
- Διάφορα υλικά, ιδιαίτερα οι ενημερώσεις και το νέο υλικό, στο κεφάλαιο για τις μελέτες περίπτωσης, ετοιμάστηκαν από τον José Ignacio Fernández Villamor και τον Boni García Gutiérrez, στις αρχές του 2007, από ένα ειδικό παρακλάδι για τις τροποποιήσεις, που έγιναν στα πλαίσια της έκδοσης εκείνου του έτους, για το μεταπτυχιακό θέμα του Joaquín Seoane στο UPM. Ένα μεγάλο κομμάτι αυτού του υλικού ενσωματώθηκε έγκαιρα για την δεύτερη έκδοση.

Περιεχόμενα

Ενότητα 1

Ελεύθερο Λογισμικό

Jesús M. González-Barahona, Joaquín Seoane Pascual and Gregorio Robles

1. Εισαγωγή
2. Λίγη ιστορία
3. Νομικές πλευρές
4. Οι Developers - Προγραμματιστές και τα κίνητρά τους
5. Οικονομία
6. Το Ελεύθερο Λογισμικό και η Δημόσια Διοίκηση
7. Engineering Ελεύθερου Λογισμικού
8. Περιβάλλοντα ανάπτυξης και τεχνολογίες
9. Μελέτες Περίπτωσης (*Case studies*)
10. Άλλοι Ελεύθεροι πόροι (*free resources*)

Ενότητα 2

Παραρτήματα

Jesús M. González Barahona, Joaquín Seoane Pascual and Gregorio Robles

1. Παράρτημα Α'. Οδηγός εκμάθησης
2. Παράρτημα Β'. Ημερομηνίες κλειδί στην ιστορία του Ελεύθερου Λογισμικού
3. Παράρτημα Γ'. Η Δημόσια Άδεια GNU (*GNU Public License*)
4. Παράρτημα Δ'. Κείμενα μερικών Σχεδίων Νόμου και σχετικά έγγραφα
5. Παράρτημα Ε'. Η άδεια “Creative Commons' Attribution-Share Alike”
6. Παράρτημα ΣΤ'. Η άδεια “GNU Free Documentation Licence”

Γλωσσάρι

ACM:	Association for Computing Machinery (Ένωση για Μηχανήματα Υπολογιστών)
AFPL:	Aladdin Free Public License (Ελεύθερη Δημόσια Άδεια Aladdin)
ALSA:	Advanced Linux Sound Architecture (Προχωρημένη Αρχιτεκτονική Ήχου του Linux)
AOL:	America Online
API:	Application program interface (Διεπαφή για πρόγραμμα εφαρμογής)
ARM:	Advanced RISC machines (Προχωρημένες μηχανές RISC)
ASCII:	American standard code for information interchange
AT&T:	American Telephone & Telegraph
AITC:	Agency of Information Technologies and Communication
ATK:	Accessibility Toolkit (Σύνολο εργαλείων προσβασιμότητας)
BIND:	Berkeley Internet Name Domain
BIRT:	Business Intelligence and Reporting Tools
BITNET:	Because It's There Network
BSA:	Business Software Alliance (Συμμαχία για Λογισμικό Επιχειρήσεων)
BSD:	Berkeley Software Distribution (Διανομή Λογισμικού του Πανεπιστ. Berkeley)
BSDI:	Berkeley Software Design Incorporated
BSI:	Bundesamt für Sicherheit in der Informationstechnik
CDDL:	Common Development and Distribution License
CD-ROM:	Compact disc read-only memory
CEPS:	Cisco Enterprise Print System (Σύστημα εκτύπωσης της Cisco Enterprise)
CERN:	Conseil Européen pour la Recherche Nucléaire (Ευρωπαϊκό Συμβούλιο για την Πυρηνική έρευνα)
CGI:	Common Gateway Interface
COCOMO:	Cost construction model (Παραμετρικό Μοντέλο υπολογισμού κόστους)
CORBA:	Common object request broker architecture
CPL:	Common Public License (Κοινή Δημόσια Άδεια)
CSRG:	Computer Systems Research Group (Ομάδα έρευνας Υπολογιστικών Συστημάτων)
CSS:	Cascading style sheet
CVS:	Control version system
DARPA:	Defence Advanced Research Projects Agency
DBUS:	Desktop Bus

DCOP:	Desktop communication protocol (Πρωτόκολλο επικοινωνίας με την επιφάνεια εργασίας)
DEC:	Digital Equipment Corporation
DECUS:	Digital Equipment Computer User Society
DFSG:	Debian Free Software Guidelines (Οδηγίες Ελεύθερου Λογισμικού Debian)
DRM:	Digital rights management (Διαχείριση Ψηφιακών Δικαιωμάτων)
DSDP:	Device Software Development Platform (Πλατφόρμα Ανάπτυξης Λογισμικού για συσκευές)
DTD:	Document type definition (Ορισμός τύπου εγγράφου)
DTP:	Data tools platform (Πλατφόρμα εργαλείων για τα δεδομένα)
DVD:	Digital video disk
ECTS:	European credit transfer scheme (Ευρωπαϊκό Σύστημα μεταφοράς credits)
EMP:	Eclipse Modelling Project
EPL:	Eclipse Public License (Δημόσια Άδεια Eclipse)
HCEST:	Higher College of Experimental Sciences and Technology (Ανώτερο Κολέγιο Πειραματικών Επιστημών και Τεχνολογίας)
ETP:	Eclipse Tools Project
FAQ:	Frequently asked questions
FDL:	Free Documentation License (Άδεια Ελεύθερων Εγγράφων)
FIC:	First International Computer
FSF:	Free Software Foundation (Ιδρυμα Ελεύθερου Λογισμικού)
FTP:	File transfer protocol (Πρωτόκολλο Μεταφοράς Αρχείων)
FUD:	Fear, uncertainty, doubt (Φ.ΑΒ.ΑΜ., Φόβος, Αβεβαιότητα, Αμφιβολία)
GCC:	Μεταγλωττιστής GNU C (<i>Compiler</i>)
GDB:	GNU Debugger (Αποσφαλματωτής GNU)
GFDL:	GNU Free Documentation License (Άδεια GNU Ελεύθερων Εγγράφων)
GIMP:	GNU Image Manipulation Program (Πρόγραμμα GNU για επεξεργασία εικόνων)
GNAT:	GNU Ada Translator
GNATS:	GNU Bug Tracking System (Σύστημα GNU εντόπισης σφαλμάτων)
GNU:	GNU's Not Unix (Το GNU δεν είναι Unix)
GPL:	General Public License (Γενική Δημόσια Άδεια)
GTK:	GIMP Toolkit (Σύνολο εργαλείων του GIMP)
GUADEC:	GNOME User and Developer European Conference (Ευρωπαϊκό Συνέδριο GNOME για χρήστες και Προγραμματιστές)
HIRD:	HURD of Interfaces Representing Depth (ΚΟΠΑΔΙ διεπαφών που αναπαριστούν το βάθος)

HTML:	Hypertext mark-up language
HTTP:	Hypertext transfer protocol (Πρωτόκολλο μεταφοράς Υπερκειμένου)
HURD:	HIRD of Unix-Replacing Daemons
R&D:	Research and development (Έρευνα & Ανάπτυξη)
IBM:	International Business Machines Corporation (Διεθνής Εταιρεία Μηχανών Γραφείου)
IDE:	Integrated development environment (Ολοκληρωμένο Περιβάλλον Ανάπτυξης)
IEC:	International Electrotechnical Commission (Διεθνής Ηλεκτροτεχνική Επιτροπή)
IETF:	Internet Engineering Task Force (Ομάδα Δράσεως Διαδικτυακού Engineering)
INRIA:	Institut National de Recherche en Informatique et en Automatique (Εθνικό Ινστιτούτο Έρευνας στην Πληροφορική και τον Αυτοματισμό)
IP:	Internet protocol (Διαδικτυακό Πρωτόκολλο)
IRC:	Internet Relay Chat
ISO:	International Standards Organization (Διεθνής Οργάνωση Προδιαγραφών)
ITU:	International Telecommunications Union (Διεθνής Ένωση Τηλεπικοινωνιών)
JDK:	Java Developer Kit (Σύνολο Εργαλείων Προγραμματισμού)
JPEG:	Joint Photographic Experts Group (Ηνωμένη Ομάδα Ειδικών Φωτογράφων)
JRE:	Java Runtime Environment
JVM:	Java Virtual Machine (Εικονική Μηχανή Java)
KBSt:	Koordinierungs-und Beratungsstelle der Bundesregierung fur Informationstechnik in der Bundesverwaltung
KDE:	K Desktop Environment (Περιβάλλον K για την επιφάνεια εργασίας)
LGPL:	Lesser General Public License (Ελάσσων Γενική Δημόσια Άδεια)
LISP:	List processing language (Γλώσσα επεξεργασίας καταλόγων)
LLC:	Limited Liability Company (Εταιρεία Περιορισμένης Ευθύνης)
IPA:	Intellectual Property Act (Νόμος για την Πνευματική Ιδιοκτησία)
LTS:	Long term support (Μακροχρόνια Υποστήριξη)
MCC:	Manchester City Council (Δημοτικό Συμβούλιο της πόλης του Μάντσεστερ)
MIT:	Massachusetts Institute of Technology (Τεχνολογικό Ινστιτούτο Μασαχουσέτης)
MPEG:	Moving Picture Experts Group (Ομάδα Ειδικών για την Κινούμενη εικόνα)
MPL:	Mozilla Public License (Δημόσια Άδεια Mozilla)

MTIC:	Mission Interministerielle de Soutien Technique pour le Developpement des technologies de l'Information et de la Communication dans l'Administration NASA (Δι- πουργική Επιτροπή για την Τεχνική Στήριξη της Ανάπτυξης τεχνολογιών της Πληροφορίας και της Ε- πικοινωνίας, εντός της Διοικήσεως της NASA)
NCSA:	National Center for Supercomputing Applications (Εθνικό Κέντρο για υπερυπολογιστικές εφαρμογές)
NPL:	Netscape Public License (Δημόσια Άδεια Netscape)
NSFNet:	National Science Foundation Network (Δίκτυο Ιδρύματος Εθνικής ερεύνης)
NUMA:	Non-uniform memory access (Μη-ομοιογενής Πρόσβαση Μνήμης)
NYU:	New York University
OASIS:	Organization for the Advancement of Structured Information Standards (Οργάνωση για την προώθηση των Προδιαγραφών Δομημένης Πληροφορίας)
ODF:	Open document format (Μορφή αρχείου Ανοικτού Εγγράφου)
ODP:	Open Directory Project (Σχέδιο Ανοικτού Φακέλλου)
OHGPL:	OpenIPCore Hardware General Public License
OLPC:	One Laptop Per Child
WTO:	World Trade Organisation (Παγκόσμιος Οργανισμός Εμπορίου)
WIPO:	World Intellectual Property Organisation (Διεθνής Οργάνωση Πνευματικής Ιδιοκτησίας)
ORB:	Object request broker (Διαχειριστής Αιτημάτων για Αντικείμενα)
OSDN:	Open Software Development Network (Δίκτυο Ανάπτυξης Ανοικτού Λογισμικού)
OSGi:	Open Services Gateway Initiative (Πρωτοβουλία Θύρας εισόδου σε Ανοικτές Υπηρεσίες)
OSI:	Open Source Initiative (Πρωτοβουλία Ανοικτού Κώδικα)
GDD:	Gross Domestic Product (Ακαθάριστο Εθνικό Προϊόν)
PDA:	Portable digital assistant (Φορητός Ψηφιακός Βοηθός)
PDF:	Portable document format (Μορφή αρχείου φορητού Εγγράφου)
PDP:	Programmed data processor (Προγραμματισμένος επεξεργαστής δεδομένων)
PHP:	PHP hypertext preprocessor (Προεπεξεργαστής Υπερκειμένου PHP)
PLOS:	Public Library of Science (Δημόσια Βιβλιοθήκη Επιστήμης)
PNG:	Portable network graphics (Φορητά Δικτυακά Γραφικά)
FAQ:	Frequently asked questions (Συχνά υποβληθείσες ερωτήσεις)
QPL:	Qt Public License (Δημόσια άδεια QT)
RCP:	Rich client platform

RDF:	Resource description framework (Framework Περιγραφής πόρων)
RFC:	Request for comments (Αίτημα για σχόλια)
RFP:	Request for proposal (Αίτημα για υποβολή προτάσεων)
RHAD:	Red Hat Advanced Development (Προχωρημένη Ανάπτυξη Red Hat)
RPM:	Red Hat Package Manager (Διαχειριστής πακέτων Red Hat)
RTF:	Rich text format (Μορφή αρχείου Εμπλουτισμένου Κειμένου)
SCO:	Santa Cruz Operation
SPE:	Secretariat of Public Education (Γραμματεία Δημόσιας Εκπαίδευσης)
SGI:	Silicon Graphics Incorporated (Εταιρεία Γραφικών Silicon)
SGML:	Standard generalised markup language
SISSL:	Sun Industry Standards Source License (Άδεια Πηγών προδιαγραφών Εταιρείας Sun)
SLS:	Softlanding Linux System
SOA:	Service oriented architecture (Αρχιτεκτονική προσανατολισμένη στην παροχή υπηρεσίας)
SPARC:	Scalable processor architecture (Αρχιτεκτονική κλιμακούμενου επεξεργαστή)
SPICE:	Simulation program with integrated circuits emphasis (Πρόγραμμα προσομοίωσης με έμφαση στα ολοκληρωμένα μικροκυκλώματα)
SSL:	Secure socket layer
TAMU:	Texas A&M University
TCP:	Transport control protocol (Πρωτόκολλο Ελέγχου Μεταφοράς)
TEI:	Text Encoding Initiative (Πρωτοβουλία Κωδικοποίησης κειμένου)
TPTP:	Test and Performance Tools Project (Σχέδιο Εργαλείων Δοκιμής και απόδοσης)
TRIPS:	Trade-related intellectual property rights (Δικαιώματα Πνευματικής Ιδιοκτησίας σχετικά με το εμπόριο)
UMTS:	Universal mobile telecommunications system
UOC:	Open University of Catalonia (Ανοικτό Πανεπιστήμιο Καταλονίας)
USA:	United States of America
USD:	United States Dollar
USENET:	User network (Δίκτυο Χρηστών)
USENIX:	Unix Users Group (Ομάδα Χρηστών Unix)
USL:	Unix System Laboratories (εργαστήρια Συστήματος Unix)
UUCP:	UNIX to UNIX copy protocol (Πρωτόκολλο αντιγραφής από Unix σε Unix)
VHDL:	Very high speed integrated circuit hardware description language (Γλώσσα περιγραφής ολοκληρωμένων κυκλωμάτων πολύ υψηλής ταχύτητας)

W3C:	World Wide Web Consortium
WIPO:	World Intellectual Property Organisation (Παγκόσμια Οργάνωση Πνευματικής Ιδιοκτησίας)
WTO:	World Trade Organisation (Παγκόσμιος Οργανισμός Εμπορίου)
WTP:	Web Tools Project (Έργο Ανάπτυξης Εργαλείων Διαδικτύου)
WWW:	World Wide Web
WYSIWYG:	What you see is what you get (“Ό,τι βλέπεις, είναι αυτό ακριβώς που παίρνεις”)
XCF:	Experimental computing facility format
XML:	Extensible markup language (Επεκτάσιμη Γλώσσα Σήμανσης)

Ελεύθερο Λογισμικό

Jesús M. González-Barahona

Joaquín Seoane Pascual

Gregorio Robles

PID_00148386



**Universitat Oberta
de Catalunya**

www.uoc.edu

Περιεχόμενα

1. Εισαγωγή	9
1.1. Η έννοια της ελευθερίας στο Λογισμικό	9
1.1.1. Ορισμός	10
1.1.2. Σχετικοί όροι	11
1.2. Κίνητρα	12
1.3. Οι συνέπειες της ελευθερίας στο λογισμικό	12
1.3.1. Για τον τελικό χρήστη	13
1.3.2. Για την Δημόσια Διοίκηση	14
1.3.3. Για τον developer	14
1.3.4. Για τον μηχανισμό ενσωμάτωσης (integrator)	15
1.3.5. Για τους παρόχους υπηρεσιών και συντήρησης	15
1.4. Περίληψη	15
2. Λίγη ιστορία	16
2.1. Το Ελεύθερο Λογισμικό πριν το Ελεύθερο Λογισμικό	16
2.2.2. Εν αρχή ήτο ελεύθερον	17
2.2.3. Η δεκαετία των 70's και οι αρχές της δεκαετίας των 80's	18
2.1.4. Η αρχική ανάπτυξη του Unix	19
2.2. Η αρχή: Το BSD, το GNU	20
2.2.1. Ο Richard Stallman, το GNU, το FSF: το Κίνημα Ελεύθερου Λογισμικού γεννιέται	20
2.2.2. Το CSRG του Berkeley	21
2.2.3. Οι απαρχές του Διαδικτύου	23
2.2.4. Άλλα projects	25
2.3. Όλα μπαίνουν στο δρόμο τους	25
2.3.1. Η αναζήτηση για έναν πυρήνα	25
2.3.2. Η οικογένεια του *BSD	26
2.3.3. Το GNU/Linux ανεβαίνει στο προσκήνιο	26
2.4. Καιρός για ωριμότητα	27

2.4.1. Το τέλος της δεκαετίας του 1990	28
--	----

2.4.2. Δεκαετία του 2000	31
--------------------------------	----

2.5. Το μέλλον: ένας αγώνας μετ' εμποδίων;	38
--	----

2.6. Περίληψη	39
---------------------	----

3. Νομικές πλευρές	40
---------------------------------	-----------

3.1. Σύντομη εισαγωγή στην πνευματική ιδιοκτησία	40
--	----

3.1.1. Copyright	41
------------------------	----

3.1.2. Εμπορικό μυστικό	42
-------------------------------	----

3.1.3. Διπλώματα ευρεσιτεχνίας (patents) και utility models	42
---	----

3.1.4. Σήματα κατατεθέντα και λογότυπα	43
--	----

3.2 Άδειες Ελεύθερου Λογισμικού	44
---------------------------------------	----

3.2.1. Είδη Αδειών	45
--------------------------	----

3.2.2. “Χαλαρές” (Permissive) άδειες	46
--	----

3.2.3. Ισχυρές άδειες	48
-----------------------------	----

3.2.4. Διανομή κάτω από περισσότερες άδειες	54
---	----

3.2.5. Γραπτή τεκμηρίωση προγράμματος	54
---	----

3.3. Περίληψη	53
---------------------	----

4. Οι Developers/ προγραμματιστές και τα κίνητρά τους	54
--	-----------

4.1. Εισαγωγή	54
---------------------	----

4.2. Ποιοί είναι developers;	55
------------------------------------	----

4.3. Τι κάνουν οι developers;	56
-------------------------------------	----

4.4. Γεωγραφική κατανομή	56
--------------------------------	----

4.5. Αφιέρωση	58
---------------------	----

4.6. Κίνητρα	59
--------------------	----

4.7. Ηγετική Ικανότητα (Leadership)	60
---	----

4.8. Περίληψη και συμπεράσματα	62
--------------------------------------	----

5. Οικονομία	62
---------------------------	-----------

5.1. Χρηματοδότηση των projects Ελεύθερου Λογισμικού	62
--	----

5.1.1. Δημόσια χρηματοδότηση	63
------------------------------------	----

5.1.2. Ιδιωτική Μη-κερδοσκοπική χρηματοδότηση	65
---	----

5.1.3. Χρηματοδότηση από όποιον επιθυμεί ορισμένες βελτιώσεις	65
---	----

5.1.4. Χρηματοδότηση με τα σχετικά οφέλη	66
--	----

5.1.5. Χρηματοδότηση ως μία εσωτερική επένδυση	67
--	----

5.1.6. Άλλοι τρόποι χρηματοδότησης	68
--	----

5.2. Επιχειρηματικά μοντέλα βασιζόμενα στο Ελεύθερο Λογισμικό	69
---	----

5.2.1. Καλύτερη γνώση	71
-----------------------------	----

5.2.2. Καλύτερη γνώση με περιορισμούς	72
---	----

5.2.3. Πηγή ενός προϊόντος Ελεύθερου Λογισμικού	73
---	----

5.2.4. Προϊόντα με πηγαίο κώδικα που έχει περιορισμούς	74
--	----

5.2.5. Ειδικές άδειες	75
-----------------------------	----

5.2.6. Πώληση εμπορικού σήματος (Brand sale)	76
--	----

5.3. Άλλες κατατάξεις επιχειρηματικών μοντέλων	76
--	----

5.3.1. Κατάταξη κατά Hecker	76
-----------------------------------	----

5.4 Επίπτωση σε μονοπωλιακές καταστάσεις	77
--	----

5.4.1. Στοιχεία που ευνοούν τα κυρίαρχα προϊόντα	77
--	----

5.4.2. Ο κόσμος του ιδιόκτητου λογισμικού	78
---	----

5.4.3. Η κατάσταση με το Ελεύθερο Λογισμικό	79
---	----

5.4.4. Στρατηγικές για δημιουργία μονοπωλίων με Ελεύθερο Λογισμικό	79
---	----

6. Ελεύθερο Λογισμικό και Δημόσια Διοίκηση	81
---	-----------

6.1. Επίπτωση στις Δημόσιες Υπηρεσίες	81
---	----

6.1.1. Πλεονεκτήματα και παράπλευρα θετικά οφέλη	82
--	----

6.1.2. Δυσκολίες υιοθέτησης και άλλα προβλήματα	85
---	----

6.2. Δράσεις των Δημοσίων Διοικήσεων στον κόσμο του Ελεύθερου Λογισμικού .	87
--	----

6.2.1. Πώς να καλυφθούν οι ανάγκες των Δημοσίων Οργανισμών;	88
---	----

6.2.2. Προώθηση της Κοινωνίας της πληροφορίας	90
---	----

6.2.3. Προώθηση της έρευνας	91
-----------------------------------	----

6.3. Παραδείγματα νομοθετικών πρωτοβουλιών	91
--	----

6.3.1. Σχέδια νόμου στη Γαλλία	92
--------------------------------------	----

6.3.2. Σχέδιο νόμου στη Βραζιλία	93
--	----

6.3.3. Σχέδια νόμου στο Περού	93
-------------------------------------	----

6.3.4. Σχέδια νόμου στη Ισπανία	94
---------------------------------------	----

7. Engineering Ελεύθερου Λογισμικού	95
--	-----------

7.1. Εισαγωγή	96
---------------------	----

7.2. Ο Καθηδρικός ναός και το bazaar	96
7.3. Ηγετική ικανότητα και η διαδικασία λήψης αποφάσεων στο bazaar	98

7.4. Διαδικασίες Ελεύθερου Λογισμικού	100
7.5. Κριτική του βιβλίου "Ο Καθηδρικός Ναός και το παζάρι"	101
7.6. Ποσοτικές μελέτες	103
7.7. Μελλοντικές δουλειές	105
7.8. Περίληψη	106

8. Τεχνολογίες και Περιβάλλοντα ανάπτυξης	106
8.1. Περιγραφή των περιβαλλόντων, των εργαλείων και των συστημάτων	107
8.2. Σχετικές γλώσσες και εργαλεία	107
8.3. Ολοκληρωμένα Περιβάλλοντα ανάπτυξης (IDE)	109
8.4. Βασικοί μηχανισμοί συνεργασίας	109
8.5. Διαχείριση Πηγών	111
8.5.1. CVS	111
8.5.2. Άλλα συστήματα διαχείρισης πηγών	115
8.6. Έγγραφο τεκμηρίωση (Documentation)	116
8.6.1. DocBook	118
8.6.2. Τα Wikis.....	118
8.7. Διαχείριση Bug και άλλα θέματα	119
8.8. Υποστήριξη για άλλες αρχιτεκτονικές	121
8.9. Ιστότοποι υποστήριξης ανάπτυξης	122
8.9.1. SourceForge	122
8.9.2. Διάδοχοι του SourceForge	124
8.9.3. Άλλοι ιστότοποι και προγράμματα	124

9. Περιπτωσιολογικές Μελέτες (Case studies)	124
9.1. Ο πυρήνας Linux	126
9.1.1. Η ιστορία του Linux	127
9.1.2. Ο τρόπος με τον οποίο δουλεύει το Linux	128
9.1.3. Η παρούσα κατάσταση του Linux	129
9.2. Το FreeBSD	131
9.2.1. Η ιστορία του FreeBSD	131
9.2.2. Η ανάπτυξη στο περιβάλλον του FreeBSD	132

9.2.3. Η διαδικασία λήψης αποφάσεων στο FreeBSD	132
---	-----

9.2.4. Οι εταιρείες με εργασίες σχετικές με το FreeBSD	133
9.2.5. Η παρούσα κατάσταση του FreeBSD	133
9.2.6. Μία ακτινογραφία του FreeBSD	134
9.2.7. Ακαδημαϊκές μελέτες στο FreeBDS	136
9.3. Το KDE	136
9.3.1. Η ιστορία του KDE	137
9.3.2. Η ανάπτυξη του KDE	137
9.3.3. Η KDE League	137
9.3.4. Η παρούσα κατάσταση του KDE	139
9.3.5. Μία ακτινογραφία του KDE	140
9.4 Το GNOME	142
9.4.1. Η ιστορία του GNOME	142
9.4.2. Το Ίδρυμα GNOME	143
9.4.3. Η βιομηχανία με εργασίες σχετικές με το GNOME	145
9.4.4. Η παρούσα κατάσταση του GNOME	146
9.4.5. Μία ακτινογραφία του GNOME	147
9.4.6. Ακαδημαϊκές μελέτες στο GNOME	148
9.5. Το Apache	149
9.5.1. Η ιστορία του Apache	149
9.5.2. Η ανάπτυξη του Apache	150
9.5.3. Μία ακτινογραφία του Apache	151
9.6. Το Mozilla	152
9.6.1. Η ιστορία του Mozilla	152
9.6.2. Μία ακτινογραφία του Mozilla	155
9.7. Το OpenOffice.org	156
9.7.1. Η ιστορία του OpenOffice.org	156
9.7.2. Η οργάνωση του OpenOffice.org	157
9.7.3. Μία ακτινογραφία του OpenOffice.org	157
9.8. Το Red Hat Linux	158
9.8.1. Η ιστορία του Red Hat	159
9.8.2. Η παρούσα κατάσταση του Red Hat.	160

9.8.3. Μία ακτινογραφία του Red Hat	160
9.9. Το Debian GNU/Linux	162
<hr/>	
9.9.1. Μία ακτινογραφία του Debian	162
9.9.2. Σύγκριση με άλλα Λειτουργικά Συστήματα	165
9.10. Το Eclipse	167
9.10.1. Η ιστορία του Eclipse	167
9.10.2. Η παρούσα κατάσταση του Eclipse	168
9.10.3. Μία ακτινογραφία του Eclipse	169
10. Άλλοι Ελεύθεροι πόροι (resources)	170
10.1. Τα σημαντικότερα Ελεύθερα resources	170
10.1.1. Επιστημονικές δημοσιεύσεις	171
10.1.2. Νόμοι και πρότυπα	171
10.1.3. Εγκυκλοπαίδειες	173
10.1.4. Σειρές μαθημάτων (Courses)	174
10.1.5. Συλλογές και Βάσεις Δεδομένων	175
10.1.6. Υλικό (Hardware)	175
10.1.7. Λογοτεχνία και Τέχνη	176
10.2. Άδειες για άλλους Ελεύθερους πόρους (resources)	176
10.2.1. Άδεια GNU free documentation	177
10.2.2. Άδειες Creative Commons	178
Βιβλιογραφία	181
 Παραρτήματα	
1. Παράρτημα Α'. Οδηγός εκμάθησης	5
2. Παράρτημα Β'. Ημερομηνίες κλειδί στην ιστορία του Ελεύθερου Λογισμικού	10
3. Παράρτημα Γ'. Η Δημόσια Άδεια GNU (GNU Public License)	22
4. Παράρτημα Δ'. Κείμενα μερικών Σχεδίων Νόμου και σχετικά έγγραφα	25
5. Παράρτημα Ε'. Η άδεια “Creative Commons' Attribution-Share Alike”	55
6. Παράρτημα ΣΤ'. Η άδεια “GNU Free Documentation Licence”	61

1. Εισαγωγή

"Αν έχεις ένα μήλο κι' εγώ, επίσης, άλλο ένα μήλο, και τα ανταλλάζουμε, τότε και οι δυο μας θα μείνουμε με ένα μήλο ο καθένας. Αλλά αν εσύ έχεις μία ιδέα και εγώ άλλη μία, και τις ανταλλάζουμε, τότε ο καθένας μας θα έχει από δύο ιδέες."

Αποδίδεται στον Bernard Shaw

Τι είναι το Ελεύθερο Λογισμικό; Τι είναι αυτό το πράγμα και ποιες είναι οι επιπτώσεις μία Αδείας Ελεύθερου Λογισμικού; Πώς αναπτύσσεται το Ελεύθερο Λογισμικό; Πώς χρηματοδοτούνται τα έργα (*projects*) Ελεύθερου Λογισμικού και ποια επιχειρηματικά μοντέλα συνδέονται μαζί τους, που τώρα βιώνουμε; Ποιο είναι το κίνητρο για τους developers, ειδικά τους εθελοντές, για να εμπλακούν σε έργα Ελεύθερου Λογισμικού; Με τι μοιάζουν αυτοί οι developers; Πώς συντονίζονται τα διάφορα έργα που τρέχουν και πώς είναι το λογισμικό που παράγουν; Κοντολογίς, ποια είναι η γενική συνολική εικόνα του Ελεύθερου Λογισμικού; Αυτές είναι οι ερωτήσεις τις οποίες θα προσπαθήσουμε να απαντήσουμε σε αυτό το βιβλίο. Ακόμη και αν το Ελεύθερο Λογισμικό αρχίζει να απασχολεί περισσότερο τα ΜΜΕ και να αποτελεί αντικείμενο συζήτησης, σε συζητήσεις αντιπαράθεσης (*debates*) μεταξύ ειδικών της Τεχνολογίας της Πληροφορικής και, έστω κι' αν το ευρύ κοινό αρχίζει να συζητά γι' αυτό, παραμένει ακόμη, εν πολλοίς, μία άγνωστη οντότητα. Ακόμη και εκείνοι που είναι πιο εξοικειωμένοι μαζί του, γνωρίζουν συχνά μόνο κάποια από τα χαρακτηριστικά του και, συνήθως, αγνοούν κάποια άλλα.

Κατ' αρχάς, σε αυτό το κεφάλαιο, θα παρουσιάσουμε τα ιδιαίτερα χαρακτηριστικά του Ελεύθερου Λογισμικού, εστιάζοντας κυρίως στην εξήγηση του υπόβαθρου, για όσους προσεγγίζουν το θέμα για πρώτη φορά, και υπογραμμίζοντας τη σημασία του. Ως μέρος αυτού του υπόβαθρου, θα αναλύσουμε τον ορισμό αυτών των όρων (για να γνωρίζουμε περί τίνος ομιλούμε) και τις βασικές επιπτώσεις της χρήσης (και της απλής ύπαρξης) του Ελεύθερου Λογισμικού.

1.1. Η έννοια της ελευθερίας στο Λογισμικό

Από τις αρχές της δεκαετίας του '70, συνηθίσαμε όλοι στο γεγονός ότι οποιοσδήποτε εμπορεύεται ένα πρόγραμμα, έχει τη δυνατότητα να επιβάλλει (και, όντως, επιβάλλει) τους όρους υπό τους οποίους το πρόγραμμα αυτό μπορεί να χρησιμοποιείται. Το να δανείσεις το πρόγραμμα σε ένα τρίτο άτομο, μπορεί, για παράδειγμα, να απαγορεύεται. Παρ' ότι το λογισμικό αποτελεί το πιο ευέλικτο και προσαρμόσιμο τεχνολογικό αγαθό που διαθέτουμε, μπορούν να επιβληθούν απαγορεύσεις (και συχνά επιβάλλονται) όσο αφορά την προσαρμογή του σε ιδιαίτερες ανάγκες, ή τη διόρθωση των σφαλμάτων του, χωρίς τη ρητή άδεια του δημιουργού του, που συνήθως διατηρεί για τον εαυτό του το αποκλειστικό δικαίωμα για αυτές τις δυνατότητες. Αλλά αυτή, είναι μόνο μία από τις δυνατότητες που η τρέχουσα νομοθεσία προσφέρει: το *Ελεύθερο Λογισμικό*, από την άλλη πλευρά, προσφέρει ελευθερίες που το *ιδιόκτητο κλειστό λογισμικό* αρνείται.

"Ιδιοταγές" (Proprietary) Λογισμικό

Σε αυτό το βιβλίο θα χρησιμοποιούμε τον όρο "*Ιδιοταγές*" (*proprietary*) Λογισμικό, για να αναφερθούμε σε οποιοδήποτε πρόγραμμα δεν μπορεί να θεωρηθεί ως Ελεύθερο, σύμφωνα με τον ορισμό που παραθέτουμε πιο κάτω.

1.1.1. Ορισμός

Έτσι, λοιπόν, ο όρος “Ελεύθερο Λογισμικό”, όπως επινοήθηκε από τον Richard Stallman στο δικό του ορισμό (Ίδρυμα Ελεύθερου Λογισμικού, “Ορισμός για το Ελεύθερο Λογισμικό” <http://www.gnu.org/philosophy/free-sw.html> [120]), αναφέρεται στις ελευθερίες που παρέχονται στον αποδέκτη του λογισμικού, οι οποίες είναι οι συγκεκριμένες τέσσερις:

- 1) Ελευθερία να τρέχουμε το πρόγραμμα σε οποιοδήποτε τόπο, για οποιονδήποτε σκοπό και πάντοτε.
- 2) Ελευθερία να μελετούμε πως δουλεύει το πρόγραμμα και να μπορούμε να το προσαρμόζουμε στις εκάστοτε ανάγκες μας. Αυτό, βέβαια, απαιτεί να έχουμε πρόσβαση στον Πηγαίο Κώδικα.
- 3) Ελευθερία να αναδιανέμουμε αντίγραφα του προγράμματος, για να μπορούμε να βοηθήσουμε τους φίλους μας και τους γείτονες.
- 4) Ελευθερία να βελτιώσουμε το πρόγραμμα και να επανασυνεισφέρουμε αυτές τις βελτιώσεις στο ευρύ κοινό. Για να γίνει αυτό, απαιτείται και πάλι ο πηγαίος κώδικας.

Ο μηχανισμός που εγγυάται αυτές τις ελευθερίες, με τρόπο συμβατό με την τρέχουσα νομοθεσία, είναι η διανομή του λογισμικού κάτω από μία ειδική άδεια, όπως θα δούμε λίγο αργότερα (Κεφάλαιο 3). Με τη βοήθεια αυτής της άδειας, ο συγγραφέας/δημιουργός δίνει στον αποδέκτη του προγράμματος την άδεια να ασκεί αυτές τις ελευθερίες, προσθέτοντας, επίσης, τους περιορισμούς που, ενδεχομένως, ο συγγραφέας θα ήθελε να ισχύουν (όπως, π.χ. την υποχρέωση να αναφέρονται οι αρχικοί δημιουργοί, σε περίπτωση αναδιανομής). Για να μπορεί η άδεια να θεωρείται ελεύθερη, αυτοί οι περιορισμοί δεν θα πρέπει να αντιβαίνουν στις προαναφερθείσες βασικές ελευθερίες.

Η αμφισημία του όρου “free” στα Αγγλικά

Ο Αγγλικός όρος “free software” περιλαμβάνει την λέξη “free”, που σημαίνει “ελεύθερο”, αλλά η ίδια λέξη μπορεί, επίσης, να σημαίνει και “δωρεάν”, πράγμα που δημιουργεί μεγάλη σύγχυση. Αυτός είναι ο λόγος για τον οποίο, σε μερικές περιπτώσεις, η Αγγλική γλώσσα δανείζεται Ισπανικές ή Γαλλικές λέξεις και, έτσι, γίνεται αναφορά σε “libre software”, κατ' αντιπαράθεση με το “Δωρεάν Λογισμικό” (Freeware).

Επομένως, οι ορισμοί για το Ελεύθερο Λογισμικό, δεν κάνουν καμία αναφορά στο ότι αυτό μπορεί να το αποκτά κανείς δωρεάν: Το Ελεύθερο Λογισμικό και το Δωρεάν Λογισμικό, είναι δύο πολύ διαφορετικά πράγματα. Αφού ξεκαθαρίσαμε τα παραπάνω, θα πρέπει και να εξηγήσουμε ότι, λόγω της τρίτης ελευθερίας, οποιοσδήποτε μπορεί να αναδιανέμει ένα πρόγραμμα, χωρίς να επιδιώκει κανένα οικονομικό όφελος ή καμιά άδεια, πράγμα που καθιστά πρακτικά αδύνατη την επίτευξη μεγάλων κερδών, απλώς αναδιανέμοντας Ελεύθερο Λογισμικό: οποιοσδήποτε κατέχει ένα Ελεύθερο Λογισμικό, μπορεί, με τη σειρά του, να το αναδιανείμει, σε μία χαμηλότερη τιμή, ή και εντελώς δωρεάν.

Σημείωση

Παρά το γεγονός ότι οποιοσδήποτε μπορεί να εμπορευθεί ένα πρόγραμμα σε οποιαδήποτε τιμή, και παρ' ότι αυτό, θεωρητικά, σημαίνει πως η τιμή αναδιανομής θα τείνει προς την οριακή τιμή της αντιγραφής του προγράμματος, υπάρχουν επιχειρηματικά μοντέλα που βασίζονται, ακριβώς, στην πώληση Ελεύθερου Λογισμικού, καθώς υπάρχουν πολλές περιπτώσεις στις οποίες ο καταναλωτής είναι διατεθειμένος να πληρώσει, παίρνοντας ως αντάλλαγμα κάποια άλλα οφέλη, όπως π.χ. μία εγγύηση, έστω και υποκειμενική, για το αποκτηθέν λογισμικό, είτε μία προστιθέμενη αξία στην επιλογή, την ενημέρωση και την οργάνωση ενός συνόλου προγραμμάτων.

Από πρακτική άποψη, διάφορα κείμενα ορίζουν με μεγαλύτερη ακρίβεια ποιες προϋποθέσεις θα πρέπει να πληροί μία άδεια για να μπορεί να θεωρείται Άδεια Ελεύθερου Λογισμικού.

Ανάμεσά τους, θα θέλαμε να αναδείξουμε, λόγω της ιστορικής του σημασίας, τον ορισμό του Ελεύθερου Λογισμικού, που έδωσε το Ίδρυμα Ελεύθερου Λογισμικού (<http://www.gnu.org/philosophy/free-sw.html>) [120], τις κατευθυντήριες γραμμές του Debian, για να κρίνουμε κατά πόσο ένα πρόγραμμα είναι ελεύθερο (http://www.debian.org/social_contract.html#guidelines) [104] και τον ορισμό του όρου “Ανοικτός Κώδικας”, από την Πρωτοβουλία για το Ανοικτό Λογισμικό (OSI, Open Source Initiative) (http://www.opensource.org/docs/definition_plain.html) [215], που είναι πολύ παρόμοιος με τους προηγούμενους.

Σημείωση

Για παράδειγμα, οι κατευθυντήριες γραμμές του Debian προχωρούν σε λεπτομέρειες, επιτρέποντας στον αρχικό δημιουργό να ζητά να μη μπορούν οι διανεμηθέντες πηγαίοι κώδικες να τροποποιούνται άμεσα, αλλά το αρχικό αντίγραφο να συνοδεύεται από χωριστά patches και τα δυαδικά προγράμματα να κυκλοφορούν με ονόματα διαφορετικά του αρχικού. Ζητούν επίσης οι Άδειες να μην “επιμολύνουν” άλλα προγράμματα που διανέμονται με τον ίδιο τρόπο.

1.1.2. Σχετικοί όροι

Ο όρος “Λογισμικό Ανοικτού Κώδικα” (Open Source Software), που προωθήθηκε από τον Eric Raymond και την Πρωτοβουλία Ανοικτού Κώδικα (Open Source Initiative) είναι ισοδύναμος με τον όρο “Ελεύθερο Λογισμικό”. Από φιλοσοφική άποψη, ο όρος είναι πολύ διαφορετικός, καθώς δίνει έμφαση στην διαθεσιμότητα του πηγαίου κώδικα και όχι στην ελευθερία του, αλλά ο ορισμός είναι πρακτικά ο ίδιος με εκείνον του Debian (“The Open Source definition”, 1998 http://www.opensource.org/docs/definition_plain.html) [183]. Αυτό το όνομα είναι, πολιτικά μιλώντας, πιο άσηπτο και δίνει έμφαση στην τεχνική πλευρά, η οποία μπορεί να προσφέρει τεχνικά οφέλη, όπως μία βελτιωμένη ανάπτυξη και επιχειρηματικά μοντέλα, καλύτερη ασφάλεια, κλπ. Δέχθηκε ισχυρή κριτική από τον Richard Stallman (“Why free software is better than open source”) [204] και το Ίδρυμα Ελεύθερου Λογισμικού (<http://www.fsf.org>) [27], και βρήκε πολύ καλύτερη απήχηση στην εμπορικό κόσμο και τις εταιρικές στρατηγικές, οι οποίες, με τον έναν ή με τον άλλο τρόπο, υποστηρίζουν αυτό το μοντέλο.

Άλλοι όροι που συνδέονται, κατά κάποιον τρόπο, με το Ελεύθερο Λογισμικό, έχουν ως ακολούθως:

Freeware	Αυτά είναι Δωρεάν Προγράμματα. Κανονικά, διανέμονται μόνον σε δυαδική μορφή (<i>binary format</i>) και μπορεί να τα αποκτήσετε χωρίς τίμημα. Ενίοτε, είναι εφικτό να έχουμε την άδεια για αναδιανομή και, άλλες φορές όχι, που σημαίνει ότι, τότε, μπορεί μόνο να το αποκτήσετε από το “επίσημο” site, που συντηρείται γι' αυτό τον σκοπό. Χρησιμοποιούνται συχνά για την προώθηση άλλων προγραμμάτων (που, συνήθως, διαθέτουν πληρέστερη λειτουργικότητα) ή υπηρεσιών. Παραδείγματα τέτοιων προγραμμάτων, περιλαμβάνουν το Skype, το Google Earth ή το Microsoft Messenger.
Shareware	Αυτά δεν είναι καν δωρεάν προγράμματα, αλλά μάλλον μία μέθοδος διανομής, αφού τα προγράμματα μπορούν, συνήθως, να αντιγραφούν δωρεάν, γενικά χωρίς τον πηγαίο κώδικα, αλλά όχι και να χρησιμοποιούνται συνεχώς χωρίς να πληρώσετε. Η απαίτηση για πληρωμή μπορεί να στηρίζεται σε μία περιορισμένη λειτουργικότητα, μέσω της αποστολής συνεχών ενοχλητικών μηνυμάτων ή της απλής επίκλησης στην ηθική του χρήστη. Επίσης, οι νομικοί όροι της αδειάς μπορεί να χρησιμοποιηθούν κατά του παραβάτη.

Charityware, careware	Αυτό είναι, συνήθως, shareware που απαιτεί η πληρωμή να κατευθύνεται προς μία επιχορηγούμενη (<i>sponsored</i>) φιλανθρωπική οργάνωση. Σε πολλές περιπτώσεις, αντί του αιτήματος πληρωμής, μπορεί να σας ζητηθεί μία εθελοντική συνεισφορά. Κάποια Ελεύθερα Λογισμικά, σαν το Vim, ζητούν εθελοντικές συνεισφορές αυτής της φύσεως (<i>Brian Molenaar, "What is the context of charityware?"</i>) [173].
Public domain	Εδώ, ο δημιουργός εκουσίως παραιτείται από όλα τα δικαιώματά του, υπέρ του public domain, και αυτό πρέπει να αναφέρεται σαφώς στο πρόγραμμα, αφού, ειδίλλως, το πρόγραμμα θα κριθεί ως ιδιόκτητο και τίποτε δεν μπορεί να κάνει κανείς με αυτό. Σε αυτή την περίπτωση, αν παρέχεται και ο πηγαίος κώδικας, τότε το πρόγραμμα είναι Ελεύθερο.
Copyleft	Αυτή είναι μία ιδιαίτερη περίπτωση Ελεύθερου Λογισμικού, όπου η άδεια απαιτεί οποιεσδήποτε διανεμόμενες τροποποιήσεις, να είναι επίσης ελεύθερες.
“Ιδιοταγές” (Proprietary), Κλειδωμένο (locked-in), Μη-Ελεύθερο (non-free)	Αυτοί είναι όροι που χρησιμοποιούνται για να αναφερθούμε σε λογισμικό που δεν είναι ούτε Ελεύθερο, ούτε Ανοικτού Κώδικα.

1.2. Κίνητρα

Όπως είδαμε, υπάρχουν δύο μεγάλες οικογένειες κινήτρων για την ανάπτυξη Ελεύθερου Λογισμικού, οι οποίες δίνουν, από ότι φαίνεται, και τα δύο ονόματα με τα οποία αυτό είναι γνωστό:

- **Τα ηθικά κίνητρα**, που υποστηρίζονται από το Ίδρυμα Ελεύθερου Λογισμικού (<http://www.fsf.org>) [27], που κληρονόμησε την κουλτούρα των hackers και υποστηρίζει τον όρο “Ελεύθερο”, επιχειρηματολογώντας ότι το λογισμικό αποτελεί γνώση, που πρέπει να μοιράζεται ανεμπόδιστα, ότι η απόκρυψη είναι κάτι το αντικοινωνικό και, επιπροσθέτως, διακηρύσσει ότι η δυνατότητα τροποποίησης προγραμμάτων συνιστά μία μορφή ελευθερίας εκφράσεως. Μπορείτε να μελετήσετε αυτά τα θέματα, σε μεγαλύτερο βάθος, διαβάζοντας “Free software, free society. Selected essays of Richard M. Stallman” [211] ή την ανάλυση του Pekka Himanen “*The hacker ethic and the spirit of the information age*”, Random House, 2001) [144].

- **Τα πραγματιστικά κίνητρα**, που υποστηρίζονται από την Πρωτοβουλία Ανοικτού Κώδικα (*Open Source Initiative/* <http://www.opensource.org>) [54], η οποία προωθεί τη χρήση του όρου Ανοικτός Κώδικας (Open Source) και υποστηρίζει την πλευρά των τεχνικο-οικονομικών πλεονεκτημάτων, που θα συζητήσουμε στο επόμενο κεφάλαιο.

Εκτός από τις δύο παραπάνω κατηγορίες κινήτρων, οι άνθρωποι που εργάζονται με το Ελεύθερο Λογισμικό, μπορεί να το πράττουν για πολλούς και διάφορους άλλους λόγους, περιλαμβανομένης και της διασκέδασης (Linus Torvalds and David Diamond, Texere, 2001) [217] ή για τα χρήματα, με επιχειρηματικά μοντέλα, εν δυνάμει, βιώσιμα. Το Κεφάλαιο 4 μελετά αυτά τα κίνητρα με λεπτομέρεια, με βάση αντικειμενικές αναλύσεις.

1.3. Οι συνέπειες της ελευθερίας του λογισμικού

Το Ελεύθερο Λογισμικό προσφέρει πολλά πλεονεκτήματα και, από τα λίγα αρνητικά, πολλά έχουν μεγαλοποιηθεί (ή τονισθεί) υπερβολικά από τους ανταγωνιστές του ιδιόκτητου λογισμικού. Το πιο βασίμιο μειονέκτημα είναι το οικονομικό, καθώς όπως είδαμε, δεν είναι δυνατόν να βγάλει κανείς πολλά λεφτά από την δημιουργία του, καθώς η διανομή μπορεί και πρέπει να γίνει από κάποιον

άλλον, διαφορετικό από τον αρχικό δημιουργό. Γι' αυτό χρειάζονται άλλα επιχειρηματικά μοντέλα και χρηματοδοτικοί μηχανισμοί, που εξετάζουμε στο κεφάλαιο 5. Άλλα μειονεκτήματα, όπως η έλλειψη υποστήριξης ή η χαμηλή ποιότητα, σχετίζονται με τη χρηματοδότηση, αλλά είναι σε πολλές περιπτώσεις ψευδή, καθώς ακόμη και το λογισμικό χωρίς καμία μορφή χρηματοδότησης, τείνει να προσφέρει ικανοποιητικά επίπεδα υποστήριξης, χάρη στα forums χρηστών και προγραμματιστών, και, συχνά, η ποιότητα είναι πολύ υψηλή.

Έχοντας κατά νου τις οικονομικές θεωρήσεις, θα πρέπει να σημειώσουμε ότι το μοντέλο κόστους του Ελεύθερου Λογισμικού είναι πολύ διαφορετικό από το μοντέλο κόστους του ιδιόκτητου λογισμικού, αφού ένα μεγάλο μέρος του αναπτύσσεται εκτός της επίσημης οικονομίας, και χρησιμοποιώντας συχνά μηχανισμούς ανταλλαγής: "σου δίνω ένα πρόγραμμα που σε ενδιαφέρει, και το προσαρμόζεις στην αρχιτεκτονική σου και κάνε τις βελτιώσεις που χρειάζεσαι." Το κεφάλαιο 7 αναλύει τους κατάλληλους μηχανισμούς Τεχνολογίας Λογισμικού (*software engineering*), για να εκμεταλλευτούμε με τον καλύτερο τρόπο αυτό το μη-αμειβόμενο ανθρώπινο δυναμικό, με τα δικά του ιδιαίτερα χαρακτηριστικά, ενώ το κεφάλαιο 8 εξετάζει τα εργαλεία που χρησιμοποιούνται για να καταστήσουν αυτή τη συνεργασία αποτελεσματική. Ένα μεγάλο μέρος του κόστους, εξ' άλλου, μειώνεται από το γεγονός ότι η πρόσβαση είναι Ελεύθερη, μιας και τα νέα λογισμικά δε χρειάζεται να ξεκινήσουν από το μηδέν, αφού μπορούν κάλλιστα να αξιοποιήσουν και να ξαναχρησιμοποιήσουν τα ήδη υπάρχοντα προγράμματα. Η διανομή τους έχει, επίσης, πολύ χαμηλότερο κόστος, καθώς γίνεται μέσω του Διαδικτύου και μέσω δωρεάν διαφήμισης σε δημόσια forums, που σχεδιάστηκαν ειδικά γι' αυτό τον σκοπό.

Ένα άλλο αποτέλεσμα των ελευθεριών αυτών είναι η τελική ποιότητα που προκύπτει από την εθελοντική συνεργασία των ανθρώπων που συνεισφέρουν ή εκείνων που εντοπίζουν και αναφέρουν άμεσα τα διάφορα bugs, σε περιβάλλοντα και καταστάσεις που είναι αδιανόητα για τον οποιονδήποτε αρχικό δημιουργό/developer. Επιπλέον, αν ένα πρόγραμμα δεν προσφέρει επαρκή ποιότητα, ο ανταγωνισμός μπορεί να το πάρει στα χέρια του και, με βάση αυτό το προϋπάρχον υλικό, να το βελτιώσει. Αυτός είναι ο τρόπος με τον οποίο η συνεργατικότητα και ο ανταγωνισμός, δύο ισχυροί μηχανισμοί, μπορούν να συνδυαστούν, για να δημιουργήσουν καλύτερη ποιότητα.

Τώρα, ας εξετάσουμε τις ευεργετικές συνέπειες για τον αποδέκτη του Ελεύθερου Λογισμικού.

1.3.1. Για τον τελικό χρήστη

Ο τελικός χρήστης, είτε είναι ένα άτομο, είτε μια εταιρεία, μπορεί να βρει πραγματικό ανταγωνισμό σε μία αγορά που χαρακτηρίζεται από μονοπωλιακές τάσεις. Για να είμαστε ακριβείς, δεν εξαρτάται απαραίτητα από την υποστήριξη του λογισμικού από την αρχική κατασκευάστρια εταιρεία, καθώς μπορεί να υπάρχουν πολυάριθμες εταιρείες, ακόμη και μικρές, με τον πηγαίο κώδικα και την τεχνογνωσία που τους επιτρέπει να κάνουν business, ενώ διατηρούν συγχρόνως και κάποια προγράμματα ελεύθερα και δωρεάν.

Η προσπάθεια να τεστάρουμε την πραγματική ποιότητα ενός προϊόντος, δεν εξαρτάται πλέον τόσο πολύ από την *αξιοπιστία* του κατασκευαστού, όσο από την καθοδήγηση που μας παρέχεται και τις ενδείξεις που προκύπτουν από το βαθμό αποδοχής της κοινότητας και τη διαθεσιμότητα του πηγαίου κώδικα. Επίσης, μπορούμε να ξεχάσουμε τις καταστάσεις του είδους "*μαύρο κουτί*", που πρέπει να εμπιστευθούμε διότι "έτσι λέμε εμείς", και μπορούμε επίσης να ξεχάσουμε την εξάρτηση από την εκάστοτε στρατηγική και τη διάθεση ενός κατασκευαστή, που μπορεί μονομερώς να αποφασίσει αν τον συμφέρει να εγκαταλείψει ή να συνεχίσει να συντηρεί ένα συγκεκριμένο προϊόν.

Η σύγκριση και αποτίμηση των προϊόντων πριν τη γενικευμένη υιοθέτησή τους, κατέστη πολύ ευκολότερη τώρα, καθώς, όλο κι' όλο που χρειάζεται να κάνουμε είναι να εγκαταστήσουμε τα εναλλακτικά προϊόντα στο πραγματικό μας περιβάλλον και να τα τεστάρουμε, ενώ, αντιθέτως, με τα ιδιόκτητα λογισμικά πρέπει να εμπιστευθούμε κάποιες εξωτερικές αναφορές ή να διαπραγματευθούμε την δυνατότητα διεξαγωγής μίας δοκιμής με τον εκάστοτε προμηθευτή, πράγμα όχι πάντα εφικτό.

Λόγω της ελευθερίας που έχουμε να τροποποιούμε το πρόγραμμα για δική μας χρήση, οι χρήστες έχουν τη δυνατότητα να το παραμετροποιήσουν ή να το προσαρμόσουν στις δικές τους απαιτήσεις, διορθώνοντας τα σφάλματα (αν υπάρχουν). Η διαδικασία αποσφαλμάτωσης (debugging) που προβλέπεται από τους χρήστες ιδιόκτητου κλειστού λογισμικού, είναι, συνήθως, εξαιρετικά εργώδης, εάν όχι σχεδόν αδύνατη, αφού, και όταν ακόμη εντοπίσουμε και διορθώσουμε τα σφάλματα, η διόρθωση αυτή θα ενσωματωθεί μόνο στην επόμενη επίσημη έκδοση του προγράμματος, η οποία μπορεί και να πάρει ολόκληρα χρόνια μέχρι να κυκλοφορήσει, και την οποία, πάνω από όλα, θα πρέπει να ξαναγοράσουμε! Με το Ελεύθερο Λογισμικό, από την άλλη πλευρά, μπορούμε να κάνουμε τις διορθώσεις μόνοι μας, εφ' όσον βέβαια κατέχουμε την τεχνογνωσία, ή διαφορετικά, μπορούμε να αναθέσουμε το έργο αυτό σε έναν εξωτερικό συνεργάτη. Μπορούμε, επίσης, τόσο άμεσα όσο και μέσω ανάθεσης σε μία εξωτερική υπηρεσία, να ενσωματώσουμε το ένα πρόγραμμα μέσα σε ένα άλλο, είτε να διατυπώσουμε μία κριτική της ποιότητάς του (π.χ. από την άποψη της ασφάλειας). Σε μεγάλο βαθμό, ο έλεγχος περνά από τον προμηθευτή στον χρήστη.

1.3.2. Για τη Δημόσια Διοίκηση

Η Δημόσια Διοίκηση αποτελεί έναν μαζικό χρήστη, με ιδιαίτερα χαρακτηριστικά, καθώς έχει μία ειδική υποχρέωση προς τους πολίτες της, είτε να παράσχει προσβάσιμες υπηρεσίες, με ουδετερότητα ως προς τους κατασκευαστές, είτε να εγγυάται την ακεραιότητα, τη χρηστικότητα, την ιδιωτικότητα και την ασφάλεια των δεδομένων τους, σε μακροπρόθεσμη βάση. Όλα τα παραπάνω, καθιστούν υποχρεωτικό για μία Δημόσια Διοίκηση να επιδεικνύει πολύ μεγαλύτερο σεβασμό και προσοχή προς τις προδιαγραφές, από ότι οι ιδιωτικές εταιρείες, και να διατηρεί τα δεδομένα σε Ανοικτά Πρότυπα (*Open Formats*) και να τα επεξεργάζεται με λογισμικά που να είναι ανεξάρτητα από τις εμπορικές στρατηγικές των- συνήθως ξένων- εταιρειών, και να είναι πιστοποιημένα ως “ασφαλή”, μέσω ενός εσωτερικού μηχανισμού αυτοελέγχου ποιότητας (internal audit). Η προσαρμογή στις προδιαγραφές αποτελεί ένα πασίγνωστο χαρακτηριστικό του Ελεύθερου Λογισμικού, πράγμα που το ιδιόκτητο κλειστό λογισμικό δεν σέβεται στον ίδιο βαθμό, διότι, απλούστατα, ενδιαφέρεται περισσότερο να δημιουργεί εγκλωβισμένους (captive) πελάτες.

Εξ' άλλου, η Δημόσια Διοίκηση, λειτουργεί και σαν παράδειγμα και οδηγός προς το ευρύτερο εταιρικό/ επιχειρηματικό περιβάλλον, εννοώντας ότι οι πρωτοβουλίες της ασκούν μία μεγάλη επιρροή, που πρέπει να κατευθύνεται στη διαμόρφωση ενός τεχνολογικού ιστού που να παραγάγει εθνικό πλούτο. Αυτός ο πλούτος μπορεί να δημιουργηθεί προωθώντας την ανάπτυξη εταιρειών που είναι αφιερωμένες στην παραγωγή νέου Ελεύθερου Λογισμικού για την Δημόσια Διοίκηση, ή με τη συντήρηση, προσαρμογή, ή την κριτική εξέταση (*auditing*) υπάρχοντος λογισμικού. Στο κεφάλαιο 6, θα εξετάσουμε αυτά τα θέματα σε περισσότερο βάθος.

1.3.3. Για τον Developer

Για τον software developer και τον παραγωγό λογισμικού, η ελευθερία αλλάζει σημαντικά τους κανόνες του παιχνιδιού. Καθιστά ευκολότερο να μπορεί κανείς να συνεχίσει να ανταγωνίζεται, ακόμη κι' αν είναι μικρός και να μπορεί να αποκτά τεχνολογία αιχμής. Μας επιτρέπει να αντλούμε όφελος από τη δουλειά των άλλων, ανταγωνιζόμενοι ακόμη και ένα άλλο προϊόν, μέσω της τροποποίησης του κώδικά του, ενώ και ο ανταγωνιστής που αντιγράψαμε μπορεί και αυτός να εκμεταλλευθεί το δικό μας κώδικα (αφού πρόκειται για *copyleft*). Αν υπάρχει σωστή διαχείριση του project, είναι δυνατόν να πετύχουμε την εκούσια συνεργασία ενός μεγάλου αριθμού ανθρώπων καθώς επίσης και την πρόσβαση σε ένα εικονικό, ελεύθερο και παγκόσμιο σύστημα διανομής. Παρ' όλα αυτά, το θέμα του πως να αποκτήσουμε πρόσβαση σε οικονομικούς πόρους, παραμένει, αν το λογισμικό δεν είναι το προϊόν μίας χρηματοδοτημένης προσπάθειας (*paid-for commission*). Το κεφάλαιο 5 ασχολείται με αυτή την πλευρά.

1.3.4. Για τον “Οργανισμό ενσωμάτωσης” (integrator)

Για τους λεγόμενους “οργανισμούς ενσωμάτωσης” (*integrators*), το Ελεύθερο Λογισμικό αποτελεί έναν παράδεισο. Σημαίνει ότι δεν υπάρχουν πια “μαύρα κουτιά” που πρέπει να ταιριάζουν μεταξύ τους, συχνά με τη χρήση αντίστροφης μηχανικής (*reverse engineering*).

Τα δύστροπα μέρη μπορούν να “στρογγυλευθούν”, ενώ διάφορα τμήματα προγραμμάτων μπορούν να συγχωνευθούν, για να παραχθεί το επιθυμητό ολοκληρωμένο προϊόν, καθώς υπάρχει μία τεράστια κοινή δεξαμενή Ελεύθερου Λογισμικού, από την οποία, αυτά τα τμήματα, μπορούν να ανασυρθούν.

1.3.5. Για τους παρόχους υπηρεσιών και συντήρησης

Η κατοχή του πηγαίου κώδικα αλλάζει τα πάντα και μας θέτει στο ίδιο επίπεδο με τον παραγωγό του λογισμικού. Αν η θέση μας δεν είναι, τελικά, ακριβώς η ίδια, αυτό συμβαίνει διότι δεν έχουμε μία εις βάθος γνώση του προγράμματος, που μόνον ο δημιουργός του διαθέτει. Αυτό σημαίνει ότι οι πάροχοι υπηρεσιών συντήρησης είναι προτιμώτερο να συμμετέχουν στα projects εκείνα που προτίθενται να συντηρήσουν. Η προστιθέμενη αξία των υπηρεσιών τους εκτιμάται πολύ περισσότερο, διότι το κόστος του προγράμματος είναι χαμηλό. Αποτελεί σαφώς την πιο ξεκάθαρη μορφή επιχειρηματικότητας με το Ελεύθερο Λογισμικό και εκείνη όπου είναι εφικτός ο μεγαλύτερος ανταγωνισμός.

1.4. Περίληψη

Το πρώτο κεφάλαιο χρησίμευσε σαν ένα είδος προκαταρκτικής γνωριμίας με τον κόσμο του Ελεύθερου Λογισμικού. Η έννοια που όρισε ο Richard Stallman, βασίζεται στις τέσσερις ελευθερίες (ελευθερία εκτέλεσης, ελευθερία μελέτης του προγράμματος, ελευθερία αναδιανομής και ελευθερία βελτιώσεων), δύο εκ των οποίων απαιτούν την πρόσβαση στον πηγαίο κώδικα. Αυτή η προσβασιμότητα και τα πλεονεκτήματά της, ενέπνευσαν και παρείχαν κίνητρο για τη σύλληψη μίας άλλης άποψης, λιγότερο βασισμένης στην ηθική και περισσότερο εστιασμένη στον πραγματισμό, την οποία υπερασπίζεται η Πρωτοβουλία Ανοικτού Κώδικα (*Open Source Initiative*), και η οποία επινόησε έναν άλλο όρο: Λογισμικό Ανοικτού Κώδικα (*Open Source Software*). Αναφέραμε, επίσης, και άλλους σχετικούς ή και αντίθετους όρους, που χρησιμεύουν για να διευκρινισθούν οι διάφορες έννοιες. Τέλος, συζητήσαμε τις συνέπειες του Ελεύθερου Λογισμικού για όλα τα εμπλεκόμενα μέρη.

2. Λίγη ιστορία

"Όταν άρχισα να εργάζομαι στο Εργαστήριο Τεχνητής Νοημοσύνης του MIT το 1971, έγινα μέλος μίας κοινότητας που μοιραζόταν το Λογισμικό και η οποία προϋπήρχε επί πολλά χρόνια. Το να μοιραζόμαστε το Λογισμικό δεν ήταν ένα χαρακτηριστικό μόνο της δικής μας κοινότητας. Ήταν κάτι τόσο παλιό, όσο και οι υπολογιστές, με τον ίδιο τρόπο που και η κοινή χρήση συνταγών μαγειρικής είναι όσο παλιά είναι και η ίδια η μαγειρική. Αλλά το κάναμε παραπάνω από οποιονδήποτε άλλο. [...] Δεν αποκαλούσαμε το Λογισμικό μας "Ελεύθερο Λογισμικό", διότι τέτοιος όρος δεν υπήρχε. Αλλά, περί αυτού ακριβώς επρόκειτο. Κάθε φορά που άλλοι επιστήμονες, προερχόμενοι από διαφορετικά Πανεπιστήμια ή εταιρείες, ήθελαν να κάνουν προσαρμογή (port) και να χρησιμοποιήσουν ένα πρόγραμμα, τους το δανείζαμε μετά χαράς. Εάν έβλεπες κάποιον να χρησιμοποιεί ένα ασυνήθιστο και ενδιαφέρον πρόγραμμα, μπορούσες, σε κάθε περίπτωση, να ζητήσεις να σε αφήσει να δεις τον πηγαίο κώδικα, για να μπορείς να τον μελετήσεις, να τον αλλάξεις, ή ακόμη και να κανιβαλίσεις κάποια τμήματά του, για να φτιάξεις ένα νέο πρόγραμμα." Richard Stallman, "The GNU Project" (originally published in the book *Open sources*) [208]

Αν και όλες οι ιστορίες σχετικές με την Τεχνολογία της Πληροφορίας (IT) είναι αναγκαστικά σύντομες, αυτή του Ελεύθερου Λογισμικού είναι μία από τις πιο μακροχρόνιες. Θα μπορούσαμε, πραγματικά, να πούμε ότι αρχικά, σχεδόν όλο το παραγόμενο λογισμικό πληρούσε τον ορισμό του Ελεύθερου Λογισμικού, ακόμη κι αν δεν υπήρχε επισήμως αυτή η έννοια. Αργότερα, η κατάσταση άλλαξε πλήρως, και το ιδιόκτητο κλειστό λογισμικό κυριάρχησε στο προσκήνιο, σχεδόν απολύτως, για μία πολύ μακρά περίοδο. Ήταν ακριβώς κατά τη διάρκεια αυτής της χρονικής περιόδου που ετέθησαν τα θεμέλια του Ελεύθερου Λογισμικού, όπως το γνωρίζουμε σήμερα και ήταν τότε που άρχισαν, λίγο-λίγο, να εμφανίζονται ελεύθερα προγράμματα. Συν τω χρόνω, αυτή η αρχή εξελίχθηκε σε μία τάση, που προόδευσε και ωρίμασε ως σήμερα, όπου το Ελεύθερο Λογισμικό συνιστά μία δυνατότητα που αξίζει να ληφθεί υπ' όψιν σε όλους, σχεδόν, τους τομείς. Η ιστορία είναι τελείως άγνωστη, τόσο που για πολλούς επαγγελματίες IT το κλειστό ιδιόκτητο λογισμικό είναι λογισμικό "στη φυσική του κατάσταση". Εν πάσει περιπτώσει, τα πράγματα έχουν μάλλον αντιστρόφως και οι σπόροι της αλλαγής που μπορούσαν, αρχικά, να εντοπισθούν στην πρώτη δεκαετία του 21^{ου} αιώνα, απέδωσαν τελικά καρπό στις αρχές της δεκαετίας του 80.

Βιβλιογραφία

Δεν υπάρχουν πολλές λεπτομερείς ιστορίες σχετικά με το Ελεύθερο Λογισμικό και, οι λίγες που υφίστανται, είναι συχνά δημοσιεύσεις που περιορίζονται στο βασικό τους θέμα. Σε κάθε περίπτωση, οι αναγνώστες που ενδιαφέρονται, μπορούν να εμβαθύνουν τη γνώση τους σχετικά με όσα αναπτύξαμε σε αυτό το κεφάλαιο, με την ανάγνωση του "*Open Source Initiative. History of the OSI*" [146] (<http://www.opensource.org/docs/history.php>), το οποίο δίνει έμφαση στην επίδραση του Ελεύθερου Λογισμικού στην επιχειρηματική κοινότητα κατά τα έτη 1998 και 1999. Επίσης, μπορούν να διαβάσουν το "*A brief history of free/open source software movement*" [190], του Chris Rasch, που καλύπτει την ιστορία του Ελεύθερου Λογισμικού έως το έτος 2000, ή το "*The origins and future of open source software*" (1999) [177], του Nathan Newman, που εστιάζει, κυρίως, στην έμμεση στήριξη του Ελεύθερου Λογισμικού ή παρόμοιων συστημάτων από την Αμερικανική Κυβέρνηση, κατά τις δεκαετίες του 1970s και του 1980.

2.1. Ελεύθερο Λογισμικό πριν το Ελεύθερο Λογισμικό

Η έννοια του Ελεύθερου Λογισμικού δεν εμφανίσθηκε παρά στις αρχές της δεκαετίας του 1980. Παρ' όλα ταύτα, η ιστορία του μπορεί να ιχνηλατηθεί μέχρι και αρκετά χρόνια νωρίτερα.

2.1.1. "Εν αρχή, ήτο ελεύθερον"

Κατά την δεκαετία του 1960, η συνολική εικόνα του κόσμου της Τεχνολογίας της πληροφορικής (IT) εκυριαρχείτο από τεράστιους υπολογιστές, που ήταν εγκατεστημένοι σε διάφορες μεγάλες

εταιρείες και σε κυβερνητικές υπηρεσίες. Ο ηγέτης των κατασκευαστών ήταν η IBM, μακράν ανώτερη των ανταγωνιστών της. Κατά τη διάρκεια εκείνης της περιόδου, όταν αγόραζε κανείς τον υπολογιστή, το λογισμικό περιλαμβάνονταν στο “πακέτο”. Για όσο καιρό ίσχυε η επί πληρωμή συνδρομή υποστήριξης, ο κατασκευαστής επέτρεπε την πρόσβαση στον κατάλογο με το δικό του λογισμικό. Επιπλέον, η ιδέα ότι τα διάφορα προγράμματα θα μπορούσαν να θεωρηθούν ως κάτι “χωριστό και ξένο”, από εμπορικής απόψεως, ήταν ασυνήθιστη.

Εκείνο τον καιρό, το software διενέμετο, συνήθως, μαζί με τον πηγαίο του κώδικα (σε πολλές, μάλιστα, περιπτώσεις, παραδιδόταν μόνον υπό μορφή πηγαίου κώδικος), και γενικότερα, χωρίς κανέναν πρόσθετο περιορισμό. Ομάδες χρηστών σαν τη SHARE (χρήστες συστημάτων της IBM) ή σαν τη DECUS (χρήστες συστημάτων της DEC) συμμετείχαν σε αυτές τις ανταλλαγές, και μέχρι ενός βαθμού, τις διοργάνωναν. Ο τομέας "*Algorithms*" του περιοδικού *Communications* της ACM, ήταν ακόμη ένα καλό παράδειγμα forum ανταλλαγών. Μπορούμε να πούμε ότι εκείνα τα πρώτα χρόνια του κόσμου του IT, το software ήταν ελεύθερο, με την έννοια ότι όποιοι είχαν πρόσβαση σε αυτό, μπορούσαν να έχουν πρόσβαση και στον πηγαίο κώδικα, και συνήθιζαν να τον μοιράζονται, να τον τροποποιούν, καθώς επίσης και να μοιράζονται και τις καινούργιες τροποποιήσεις.

Στις 30 Ιουνίου του 1969, η IBM ανακοίνωσε ότι από το 1970, θα άρχιζε να πουλάει χωριστά ένα μέρος του λογισμικού της (Burton Grad, 2002) [131]. Αυτό, βέβαια, σήμαινε ότι τα προγράμματα που χρειαζόταν οι πελάτες της, δεν θα συμπεριλαμβανόταν, πλέον, μέσα στην τιμή του hardware. Ήταν τότε που το Software άρχισε να γίνεται αντιληπτό σαν κάτι με μία ενδογενή δική του αξία και, κατά συνέπεια, έγινε όλο και πιο συχνό να βλέπει κανείς τον αυστηρό περιορισμό της πρόσβασης στα προγράμματα, οπότε και η δυνατότητα των χρηστών να μοιράζονται, να μπορούν να τροποποιήσουν ή και να μελετούν το software, συρρικνώθηκε όσο περισσότερο γινόταν (τόσο από τεχνικής όσο και από νομικής απόψεως). Με άλλα λόγια, η κατάσταση μετεξελίχθηκε σε κάτι που συνεχίζει να ισχύει στον κόσμο του Λογισμικού ως τις αρχές του 21^{ου} αιώνα.

Βιβλιογραφία

Οι αναγνώστες που ενδιαφέρονται να μάθουν γι' αυτή τη μεταβατική περίοδο, μπορούν να διαβάσουν πχ. το "How the ICP Directory began" [226] (1998), όπου ο Larry Welke συζητά πως προέκυψαν οι πρώτοι κατάλογοι λογισμικού που δεν συνδεόταν με κανέναν συγκεκριμένο κατασκευαστή και πως, κατά τη διάρκεια αυτής της φάσης, ανακαλύφθηκε πως οι Εταιρείες ήταν διατεθειμένες να πληρώσουν για προγράμματα που δεν ήταν φτιαγμένα από τον κατασκευαστή των υπολογιστών τους.

Στα μισά της δεκαετίας του 1970, το να βρει κανείς ιδιόκτητο λογισμικό, ήταν ήδη κοινό φαινόμενο στον χώρο του IT. Αυτό σηματοδότησε μία τεράστια αλλαγή νοοτροπίας ανάμεσα στους επαγγελματίες που δούλευαν με το software και αποτέλεσε την αρχή της άνθησης ενός μεγάλου αριθμού εταιρειών που αφιερωνόταν σε αυτή την νέα μορφή επιχειρηματικότητας. Έπρεπε να περάσει, σχεδόν, ακόμη μία δεκαετία, πριν εμφανισθεί αυτό που γνωρίζουμε ως Ελεύθερο Λογισμικό, υπό μία οργανωμένη μορφή και ως αντίδραση στην κατάσταση αυτή.

2.1.2. Η δεκαετία του '70 και οι αρχές της δεκαετίας του '80

Ακόμη και όταν η υπερισχύουσα τάση ήταν υπέρ της διερεύνησης του μοντέλου του ιδιόκτητου λογισμικού, υπήρχαν και τότε πρωτοβουλίες που διέθεταν μερικά από τα χαρακτηριστικά που αργότερα θα θεωρούσαμε ως “Ελεύθερο Λογισμικό”. Σε μερικές από αυτές τις περιπτώσεις, υπήρχε, όντως, παραγωγή Ελεύθερου Λογισμικού, ακριβώς με τον ίδιο τρόπο που θα το ορίζαμε και σήμερα. Από αυτές, θα μνημονεύσουμε την SPICE, την TeX και το Unix, που είναι μία πολύ πιο περίπλοκη περίπτωση.

Το SPICE (*Simulation Program with Integrated Circuit Emphasis*), είναι ένα πρόγραμμα που ανεπτύχθη από το Πανεπιστήμιο της California, στο Berkeley, για να μπορέσει να προσομοιώσει τα ηλεκτρικά χαρακτηριστικά ενός ολοκληρωμένου κυκλώματος. Ανεπτύχθη και ετέθη σε δημόσια χρήση (*public domain*) από τον ίδιο του τον δημιουργό, τον Donald O. Pederson, το 1973. Η SPICE ήταν, αρχικά, ένα εργαλείο διδασκαλίας και, ως τέτοιο, διαδόθηκε γρήγορα σε διεθνές διαπανεπιστημιακό επίπεδο. Εκεί, χρησιμοποιήθηκε από πολλούς φοιτητές, στα πλαίσια της μόλις τότε αναδυόμενης ειδικότητας: σχεδίαση ολοκληρωμένων κυκλωμάτων. Καθώς βρισκόταν σε δημόσια χρήση (*public domain*), το SPICE μπορούσε να διανεμηθεί, να τροποποιηθεί, να μελετηθεί. Μπορούσε ακόμη και να προσαρμοσθεί σε ιδιαίτερες απαιτήσεις και, εκείνη η έκδοση, μπορούσε, μετά, να πωλείται σαν ιδιοταγές προϊόν (πράγμα που έκαναν πολυάριθμες εταιρείες σε άπειρες περιστάσεις, κατά τη διάρκεια της διαδρομής τους). Με αυτά τα χαρακτηριστικά, το SPICE είχε όλα τα χαρτιά για να γίνει το πρότυπο της βιομηχανίας (*industry standard*), με τις διάφορες εκδόσεις του. Και, όντως, αυτό ακριβώς συνέβη. Επρόκειτο, πιθανώς, για το πρώτο πρόγραμμα με χαρακτηριστικά ελεύθερου λογισμικού, που για κάποιο διάστημα κυρίευσε την αγορά των προσομοιωτών για ολοκληρωμένα κυκλώματα και, το οποίο, αναμφίβολα, τα κατάφερε διότι είχε αυτά τα συγκεκριμένα χαρακτηριστικά (πέραν των αδιαμφισβήτητων τεχνικών προτερημάτων).

Βιβλιογραφία

Μπορείτε να μάθετε παραπάνω για την ιστορία του SPICE, εάν συμβουλευθείτε το "The life of SPICE", που παρουσιάστηκε κατά το *Bipolar Circuits and Technology Meeting*, Minneapolis, MN, USA, τον Σεπτέμβριο του 1996 [175]. Μπορείτε επίσης να βρείτε την ιστοσελίδα του SPICE στη διεύθυνση: <http://bwrc.eecs.berkeley.edu/Courses/IcBook/SPICE/>.

Ο Donald Knuth άρχισε να αναπτύσσει το TeX, κατά την διάρκεια ενός έτους εκπαιδευτικής αδείας (*sabbatical*), το 1978. Το TeX είναι ένα σύστημα ηλεκτρονικής τυπογραφίας που χρησιμοποιείται συνήθως για την παραγωγή εγγράφων υψηλής ποιότητας. Από την αρχή, ο Knuth χρησιμοποίησε μία άδεια που σήμερα θα θεωρείτο ως άδεια ελεύθερου Λογισμικού. Όταν το σύστημα κρίθηκε ως επαρκώς σταθερό, το 1985, εκείνος διατήρησε αυτή την άδεια. Σε αυτό το σημείο, το TeX ήταν από τα μεγαλύτερα και πιο γνωστά συστήματα που θα μπορούσαν να θεωρηθούν ως Ελεύθερο Λογισμικό.

Βιβλιογραφία

Μπορείτε να βρείτε μερικά από βασικά στάδια ανάπτυξης στην ιστορία του TeX, αν ανατρέξετε στο <http://www.math.utah.edu/software/plot79/tex/history.html> [39]. Για παραπάνω λεπτομέρειες, το αντίστοιχο άρθρο της Wikipedia είναι, επίσης, εξαιρετικά χρήσιμο: <http://www.wikipedia.org/wiki/TeX>[233].

2.1.3. Η αρχική ανάπτυξη του Unix

Το Unix, ένα από τα πρώτα φορητά λειτουργικά συστήματα, δημιουργήθηκε αρχικά από τους Thompson και Ritchie (μεταξύ άλλων), στο περιβάλλον των εργαστηρίων της AT&T's Bell Labs. Μετά τη γέννησή του περί το 1972, συνέχισε να αναπτύσσεται, οδηγώντας στην εμφάνιση αναρίθμητων ποικιλιών του, που επωλούντο (στην κυριολεξία) από δεκάδες εταιρείες.

Κατά τα έτη 1973 και 1974, το Unix έφθασε σε διάφορα Πανεπιστήμια και ερευνητικά κέντρα παγκοσμίως, με μία άδεια που επέτρεπε την χρήση του για ακαδημαϊκούς σκοπούς. Αν και υπήρχαν ορισμένοι περιορισμοί που δεν επέτρεπαν την ελεύθερη διάδοσή του μεταξύ των εταιρειών που είχαν μία άδεια, η λειτουργία του ήταν πολύ παρόμοια με εκείνη που θα εσυναντάτο αργότερα σε πολλές κοινότητες ελεύθερου λογισμικού.

Εκείνοι που είχαν πρόσβαση στον πηγαίο κώδικα του Unix, είχαν να κάνουν με ένα σύστημα που

μπορούσαν να μελετήσουν, να βελτιώσουν και να επεκτείνουν. Μία ολόκληρη κοινότητα προγραμματιστών/ developers ανεπτύχθη τριγύρω του, η οποία σύντομα βρήκε το κέντρο βάρους της γύρω από το CSRG του Πανεπιστημίου της California, στο Berkeley. Αυτή η κοινότητα ανέπτυξε τη δική της κουλτούρα, που όπως θα δούμε αργότερα, υπήρξε πολύ σημαντική στην ιστορία του ελεύθερου λογισμικού. Το Unix απέτελεσε, σε κάποιο βαθμό, μία πρώιμη δοκιμή για το τι θα βλέπαμε με το GNU και το Linux, αρκετά χρόνια αργότερα. Περιοριζόταν σε μία πολύ μικρότερη κοινότητα και η άδεια AT&T ήταν απαραίτητη, αλλά, ως προς όλα τα υπόλοιπα, η ανάπτυξή του ήταν πολύ παρόμοια (σε ένα πολύ λιγότερο διασυνδεδεμένο κόσμο).

Μέθοδοι ανάπτυξης χαρακτηριστικές του Ελεύθερου Λογισμικού

Από το Netizens. Μπορούμε να διαβάσουμε λίγα πράγματα σχετικά με την ιστορία και την επίδραση του Usenet και του Internet (*IEEE Computer Society Press*, 1997 [139], σελ. 139), που θα μπορούσαν να ισχύουν και στην περίπτωση πολλών άλλων projects Ελεύθερου Λογισμικού: "Το γεγονός ότι ο πηγαίος κώδικας ήταν ανοικτός και διαθέσιμος, συνεισέφερε στην αξία που απέκτησε το Unix, κατά τα πρώιμα στάδια της ανάπτυξής του. Μπορούσε να εξετασθεί, να βελτιωθεί και να παραμετροποιηθεί".

Στη σελ. 142 του ίδιου έργου, αναφέρονται τα ακόλουθα: "Οι πρωτοπόροι, όπως ο Henry Spencer, συμφωνούν για το πόσο σημαντική ήταν η διαθεσιμότητα του πηγαίου κώδικα για όλους εκείνους που συμμετείχαν στην κοινότητα Unix. Σημειώνει, πως η κατοχή όλων των πηγαίων κωδίκων, κατέστησε εφικτό τον εντοπισμό και διόρθωση των bugs που είχαν ανακαλύψει. [...] Ακόμη και προς τα τέλη της δεκαετίας του 1970 και τις αρχές της δεκαετίας του 1980, πρακτικά, κάθε ιστοσελίδα σχετική με το Unix, διέθετε τους πλήρεις κώδικες".

Το κείμενο του Marc Rochkind "*Συνέντευξη με τον Dick Haight*", είναι ακόμη πιο ξεκάθαρο (*Unix Review*, May 1986) [198]: "αυτό ήταν ένα από τα μεγάλα πλεονεκτήματα του Unix, στα πρώιμα στάδια: οι άνθρωποι, πραγματικά, μοιραζόταν το υλικό τους, ο ένας με τον άλλον. [...] Από την κοινή χρήση του υλικού, όχι μόνο κατανοήσαμε ένα σωρό πράγματα σε εκείνους τους παλιούς καιρούς, αλλά ούτε και χρειαζόταν ποτέ να ανησυχήσουμε για το πως ακριβώς λειτουργούσαν τα προγράμματα, αφού είχαμε πάντα την δυνατότητα να πάμε κατ' ευθείαν στην πηγή."

Με τον χρόνο, και το ίδιο το Unix κατέστη ένα πρώιμο παράδειγμα των προβλημάτων που θα μπορούσαν να ανακύψουν από τα ιδιόκτητα συστήματα που, σε μία πρώτη ματιά, "είχαν κάποια χαρακτηριστικά Ελεύθερου Λογισμικού".

Κατά τα τέλη της δεκαετίας του 1970 και, ιδιαίτερα, κατά την διάρκεια της δεκαετίας του 1980, η εταιρεία AT&T άλλαξε την πολιτική της και κατέστησε την πρόσβαση στις νέες εκδόσεις του Unix δύσκολη και ακριβή. Η φιλοσοφία των πρώτων ημερών που ήταν εκείνη που έκανε το Unix τόσο δημοφιλές ανάμεσα στους developers, άλλαξε ριζικά σε τέτοιο βαθμό, που το 1991 η AT&T έφθασε ακόμη και να υποβάλλει μήνυση στο Πανεπιστήμιο του Berkeley, διότι δημοσίευσε τον κώδικα του Unix BSD, που είχε αναπτύξει η ομάδα Berkeley's CSRG. Αλλά αυτή είναι μία άλλη ιστορία που θα ανοίξουμε αργότερα.

2.2. Η αρχή: το BSD και το GNU

Όλες οι περιπτώσεις που συζητήσαμε στα προηγούμενα κεφάλαια, ήταν είτε ατομικές πρωτοβουλίες, είτε δεν ικανοποιούσαν, με την στενή έννοια, τις προδιαγραφές του Ελεύθερου Λογισμικού. Δεν ήταν παρά μόνον στις αρχές της δεκαετίας του 1980 που εμφανίσθηκαν τα πρώτα οργανωμένα και ενσυνείδητα projects για τη δημιουργία συστημάτων που να περιλαμβάνουν Ελεύθερο Λογισμικό. Κατά τη διάρκεια αυτής της περιόδου, άρχισαν να τίθενται τα θεμέλια για τις ηθικές, τις νομικές και τις οικονομικές βάσεις αυτών των projects (ίσως ακόμη σημαντικότερο), με την ανάπτυξή τους και την ολοκλήρωσή τους έως και τις μέρες μας. Και αφού το νέο φαινόμενο χρειαζόταν ένα όνομα, τότε ήταν που, για πρώτη φορά, επινοήθηκε ο όρος *Ελεύθερο Λογισμικό*.

2.2.1. Ο Richard Stallman, το GNU, το FSF: γέννηση του Κινήματος Ελεύθερου Λογισμικού

Στις αρχές του 1984, ο Richard Stallman, που εκείνη την εποχή εργαζόταν στο MIT AI Lab, παραιτήθηκε από τη θέση του, για να αρχίσει να δουλεύει στο GNU project. Ο Stallman θεωρούσε εαυτόν ως έναν *hacker*, από εκείνους που απολαμβάνουν να μοιράζονται τα τεχνολογικά τους ενδιαφέροντα και τον κώδικά τους. Δεν του άρεσε ο τρόπος με τον οποίο η άρνησή του να υπογράψει συμφωνίες αποκλειστικότητας ή μη-μοιρασιάς, τον καθιστούσε αποκλεισμένο μέσα

στον ίδιο του τον κόσμο, ούτε πως η χρήση του ιδιόκτητου λογισμικού μέσα στο περιβάλλον του, τον άφηνε ανήμπορο μπροστά σε καταστάσεις που μπορούσαν, εύκολα, να είχαν επιλυθεί νωρίτερα.

Η ιδέα του, όταν εγκατέλειψε το MIT, ήταν να φτιάξει ένα πλήρες λειτουργικό σύστημα, για γενική χρήση, αλλά εντελώς Ελεύθερο ("*The GNU Project*", DiBona et al.) [208]. Το σύστημα (και το project που είχε την ευθύνη της πραγμάτωσής του) ονομάστηκε GNU ("*GNU's not Unix*", ένα αυτοανακυκλούμενο -"recursive" ακρώνυμο). Αν και από την αρχή το GNU project περιέλαβε λογισμικό που ήταν ήδη διαθέσιμο (όπως το TeX ή, αργότερα, το σύστημα X Window), υπήρχαν ακόμη πολλά να φτιαχτούν. Ο Richard Stallman άρχισε να γράφει έναν μεταγλωττιστή C (τον *GCC compiler*) και έναν editor (τον *Emacs*), αμφότεροι εκ των οποίων είναι σε χρήση (και πολύ δημοφιλείς) ακόμη και σήμερα.

Από την αρχή του GNU project, ο Richard Stallman ανησυχούσε για τις ελευθερίες που θα είχαν οι χρήστες του λογισμικού. Δεν ήθελε να συνεχίζουν να απολαμβάνουν τα ίδια δικαιώματα (τροποποίηση, αναδιανομή, κλπ.) μόνον εκείνοι που λάμβαναν προγράμματα κατ' ευθείαν από το GNU project, αλλά και εκείνοι που τα έπαιρναν μετά από έναν ολόκληρο κύκλο αναδιανομών και (ενδεχομένως) τροποποιήσεων. Γι' αυτό το λόγο, σχεδίασε την άδεια GPL, που ήταν, ίσως, η πρώτη άδεια λογισμικού που σχεδιάστηκε ειδικά για να εγγυάται ότι ένα πρόγραμμα θα παρέμενε, με αυτό τον τρόπο, ελεύθερο. Αυτό τον γενικό μηχανισμό που χρησιμοποιούν οι άδειες τύπου GPL για να διασφαλίσουν αυτές τις εγγυήσεις, ο Richard Stallman τον ονόμασε "*copyleft*", και αυτό συνεχίζει να παραμένει και το όνομα μίας μεγάλης οικογένειας αδειών Ελεύθερου Λογισμικού (Ίδρυμα Ελεύθερου Λογισμικού, GNU General Public Licence, έκδοση 2, Ιούνιος 1991) [118].

Ο Richard Stallman ίδρυσε, επίσης, το Ίδρυμα Ελεύθερου Λογισμικού (Free Software Foundation - FSF), για να προσελκύσει χρηματοδότηση, η οποία του χρειάζεται για να αναπτύσσει και να προστατεύει το Ελεύθερο Λογισμικό, και καθόρισε και τις ηθικές του αρχές, μέσα από το "GNU Manifesto" (Free Software Foundation, 1985) [117] και το "*Γιατί το Λογισμικό δεν θα έπρεπε να έχει ιδιοκτήτες - Why software should not have owners*" (Richard Stallman, 1998) [207].

Από τεχνικής απόψεως, το GNU project στήθηκε ως μία εξαιρετικά δομημένη προσπάθεια, με πολύ ξεκάθαρους στόχους. Η συνήθης μεθοδολογία βασιζόταν σε σχετικά μικρές ομάδες ανθρώπων (συχνά εθελοντών), οι οποίοι θα ανέπτυσσαν διάφορα εργαλεία, που στη συνέχεια θα "δέναν" όλα αρμονικά στο μεγάλο παζλ (το σύστημα GNU).

Η δομή σε αρθρώματα (*modules*) του Unix, από την οποία αυτό το project εμπνεύστηκε, συνέπεσε πλήρως με εκείνη την ιδέα. Η μέθοδος εργασίας, γενικά, συμπεριελάμβανε τη χρήση του Διαδικτύου, αλλά καθώς αυτό δεν ήταν και τόσο διαδεδομένο την εποχή εκείνη, το Ίδρυμα Ελεύθερου Λογισμικού άρχισε να πουλά ταινίες, επί των οποίων "έγραφε" τα διάφορα προγράμματα. Αυτό σημαίνει ότι ήταν, πιθανώς, μία από τις πρώτες οργανώσεις που πέτυχαν μία οικονομικο-λογιστική ισορροπία (αν και με έναν περιορισμένο τρόπο), από την πώληση Ελεύθερου Λογισμικού.

Στην αρχή της δεκαετίας του 1990, περίπου 6 χρόνια μετά την ίδρυση του όλου project, το GNU πλησίασε πολύ κοντά στην επίτευξη της δημιουργίας ενός πλήρους συστήματος, παρόμοιου με το Unix. Εν πάσει περιπτώσει, σε εκείνη τη φάση, δεν είχε ακόμη πετύχει να φτιάξει ένα από τα κρίσιμα στοιχεία: τον *πυρήνα* του συστήματος (γνωστού και ως "*kernel*"), το τμήμα εκείνο του Λειτουργικού Συστήματος που διαχειρίζεται το hardware και επιτρέπει στις διάφορες επί μέρους εφαρμογές να μοιράζονται πόρους και, κατ' ουσίαν, να λειτουργούν.

Πάντως, το λογισμικό GNU ήταν πολύ δημοφιλές μεταξύ των χρηστών των διαφόρων ποικιλιών

του Unix, που ήταν και το συχνότερα χρησιμοποιούμενο λειτουργικό σύστημα στις επιχειρήσεις, τον καιρό εκείνο. Επί πλέον, το GNU project είχε καταφέρει να γίνει αρκετά γνωστό στους επαγγελματίες του IT και, ιδιαίτερα, σε εκείνους που εργαζόταν σε Πανεπιστήμια. Εκείνη την χρονική περίοδο, τα προϊόντα του είχαν κατακτήσει μία καταξιωμένη φήμη για τη σταθερότητά τους και την καλή ποιότητα.

2.2.2. To CSRG του Berkeley

Από το 1973, το CSRG (*Computer Science Research Group*) του Πανεπιστημίου της Καλιφόρνιας, στο Berkeley, υπήρξε ένα από τα κέντρα όπου έλαβε χώρα η ανάπτυξη των περισσότερων σχετικών με το Unix σχεδίων, ιδιαίτερα ανάμεσα στο 1979 και το 1980. Δεν κάνανε μόνο προσαρμογή (*port*) πολλών εφαρμογών και δεν έφτιαχναν απλώς καινούργια προγράμματα που τρέχανε από την αρχή σε περιβάλλον Unix, αλλά επέφεραν και σημαντικές βελτιώσεις στον πυρήνα, με προσθήκη μεγάλης λειτουργικότητας. Για παράδειγμα, κατά τη δεκαετία του 80, διάφορες συμβάσεις DARPA (από το Αμερικανικό Υπουργείο Αμύνης) συνεισέφεραν χρηματοδότηση για την εφαρμογή του πρωτοκόλλου TCP/ IP, το οποίο θεωρείται, ακόμη και μέχρι σήμερα, το σημείο αναφοράς για τα πρωτόκολλα που κάνουν όλο το Διαδίκτυο να δουλεύει (στη διαδικασία αυτή, επήλθε μία αλληλεξάρτηση ανάμεσα στην ανάπτυξη του Internet και την επέκταση των σταθμών εργασίας σε περιβάλλον *Unix*). Πολλές εταιρείες χρησιμοποίησαν τις επιτυχίες του CSRG ως βάση για δικές τους εκδόσεις Unix, δημιουργώντας διάφορα γνωστά- τότε- συστήματα, όπως το SunOS (της *Sun Microsystems*) ή το Ultrix (της *Digital Equipment*). Αυτός ήταν ο τρόπος με τον οποίο το Berkeley κατέστη μία από τις δύο βασικές πηγές του Unix, μαζί με την λεγόμενη “επίσημη”, δηλ. την AT&T.

Για να μπορεί κανείς να κάνει χρήση όλου του κώδικα που είχε παραχθεί από το CSRG (καθώς και του κώδικα που είχαν συνεισφέρει όλοι οι συνεργάτες της κοινότητας Unix και οι οποίοι έπαιζαν- σε κάποιο βαθμό- τον ρόλο του συντονιστή), ήταν απαραίτητο να έχει στην κατοχή του την άδεια της AT&T για το Unix . Αυτή όμως ήταν όλο και πιο δύσκολο να την έχει κανείς (πέραν του ότι ήταν όλο και ακριβότερη), ειδικά εάν ήταν αναγκαία η πρόσβαση στον πηγαίο κώδικα του συστήματος. Σε μία μερική προσπάθεια να υπερκερασθεί αυτό το πρόβλημα, το CSRG εξέδωσε, τον Ιούνιο του 1989, ένα τμήμα του Unix που συνδεόταν με το TCP/IP (η εφαρμογή των πρωτοκόλλων εντός του πυρήνα και τα σχετικά *utilities*), που όμως δεν περιέκλειε τον κώδικα της AT&T code. Αυτό ονομάστηκε “*Networking Release 1*” (Net-1). Η άδεια υπό την οποία εξεδόθη ήταν η περίφημη άδεια BSD και η οποία, με την εξαίρεση κάποιων προβλημάτων με τις ρήτρες που αφορούσαν τις υποχρεώσεις διαφήμισης, εθεωρείτο πάντα σαν ένα παράδειγμα μίας μινιμαλιστικής άδειας Ελεύθερου Λογισμικού (που, εκτός από το να επιτρέπει την ελεύθερη αναδιανομή, επιτρέπει επίσης και την ενσωμάτωση στο πλαίσιο εμπορικών προϊόντων). Επιπρόσθετα, το CSRG δοκίμασε ένα νέο μοντέλο χρηματοδότησης (που ήδη δοκιμαζόταν με επιτυχία από το FSF): άρχισε την πώληση ταινιών με την δική του διανομή, έναντι USD 1,000 εκάστη. Παρά το γεγονός ότι θα μπορούσε, πλέον, ο καθένας να αναδιανέμει το περιεχόμενο των ταινιών αυτών, χωρίς να αντιμετωπίσει κανένα πρόβλημα (διότι η ίδια η άδεια το επέτρεπε), το CSRG κατάφερε να πωλήσει ταινίες σε χιλιάδες επιχειρήσεων, προσπορίζόμενο έτσι κεφάλαια με τα οποία μπόρεσε να συνεχίσει να αναπτύσσεται.

Έχοντας διαπιστώσει την επιτυχία της διανομής Net-1, ο Keith Bostic πρότεινε να ξαναγραφεί όλος ο κώδικας που απέμενε από το αρχικό Unix της AT&T. Παρά τον σκεπτικισμό ορισμένων μελών του CSRG, απηύθυνε μία δημόσια έκκληση, ζητώντας βοήθεια για την αποπεράτωση του εγχειρήματος και, λίγο-λίγο τα *utilities* (που ξαναγράφηκαν με βάση τις προδιαγραφές) ενσωματώθηκαν στο σύστημα του Berkeley. Στο εντωμεταξύ, η ίδια διαδικασία έλαβε χώρα και για τον πυρήνα, με ένα τέτοιο τρόπο ώστε ο περισσότερος κώδικας που δεν είχε παραχθεί από το Berkeley ή από τους εθελοντές συνεργάτες, ξαναγράφηκε ανεξαρτήτως. Τον Ιούνιο του 1991, αφού ελήφθη η άδεια από το Δ.Σ. του Πανεπιστημίου του Berkeley, δόθηκε στο κοινό η διανομή

Networking Release 2 (*Net-2*), με σχεδόν όλον τον κώδικα του πυρήνα και όλα τα utilities που συνθέταν ένα πλήρες σύστημα Unix. Το όλο “πακέτο” διανεμήθη ξανά με την άδεια BSD και πουλήθηκαν χιλιάδες ταινιών, με τιμή USD 1,000 ανά μονάδα.

Μόλις 6 μήνες μετά την δημοσίευση του *Net-2*, ο Bill Jolitz έγραψε και το υπόλοιπο κομμάτι κώδικα που έλειπε από τον πυρήνα, για να λειτουργήσει στα πλαίσια της δομής του i386, εκδίδοντας έτσι το 386BSD, το οποίο και διανεμήθη μέσω του Διαδικτύου. Με βάση αυτόν τον κώδικα, προέκυψαν αργότερα, κατά σειρά, όλες οι επόμενες εκδόσεις των συστημάτων της οικογένειας *BSD: πρώτα εμφανίστηκε το NetBSD, ως μία συλλογή μικρών διορθώσεων (patches), που είχαν συνεισφέρει διάφοροι μέσω Διαδικτύου, για τη βελτίωση του 386BSD. Αργότερα, εμφανίστηκε το FreeBSD, ως μία προσπάθεια εστίασης στην υποστήριξη της αρχιτεκτονικής του i386. Πολλά χρόνια αργότερα, γεννήθηκε το OpenBSD project, με έμφαση σε θέματα ασφάλειας. Υπήρξε επίσης μία ιδιοταγής έκδοση, βασισμένη στο *Net-2* (αν και ήταν, οπωσδήποτε, πρωτότυπη, καθώς προσέφερε στους πελάτες της όλο τον πηγαίο κώδικα, ως μέρος της βασικής διανομής), που δημιουργήθηκε ανεξάρτητα, από την εταιρεία BSDI (*Berkeley Software Design Inc.*) που δεν υπάρχει πλέον.

Η Unix System Laboratories (*USL*), θυγατρική της AT&T, με την άδεια του Unix υπό την κατοχή της, προσπάθησε να εγείρει αγωγή, κατ' αρχάς εναντίον της BSDI και στη συνέχεια, κατά του Πανεπιστημίου της Καλιφόρνιας, εν μέρει ως αντίδραση προς την διανομή που είχε εκδώσει η BSDI. Η κατηγορία συνίστατο στο ότι η BSDI είχε διανείμει την δική της πνευματική ιδιοκτησία χωρίς άδεια. Μετά από διάφορους νομικούς ελιγμούς (που συμπεριελάμβαναν και μία αντιμήνυση, εκ μέρους του Πανεπιστημίου της Καλιφόρνιας, κατά της USL), η Novell εξαγόρασε τα δικαιώματα για το Unix από την USL και τον Ιανουάριο του 1994 επετεύχθη ένας εξωδικαστικός συμβιβασμός με το Πανεπιστήμιο της Καλιφόρνιας. Ως αποτέλεσμα αυτού του συμβιβασμού, η CSRG διένειμε την έκδοση 4.4BSD-Lite, που σύντομα χρησιμοποιήθηκε από όλα τα projects της οικογένειας του *BSD. Συντόμως μετά (μετά την έκδοση *4.4BSD-Lite Release 2*), η CSRG εξαφανίστηκε. Σε αυτό το σημείο, πολλοί φοβήθηκαν ότι ίσως να είχε σημάνει το τέλος των συστημάτων *BSD, αλλά ο χρόνος απέδειξε ότι παραμένουν ακόμη γεμάτα σφρίγος και ζωντάνια, υπό ένα νέο management που είναι πιο τυπικό για τα projects Ελεύθερου Λογισμικού. Ακόμη και κατά την πρώτη δεκαετία του 2000, τα projects που διαχειρίζεται η οικογένεια *BSD είναι από τα παλιότερα και τα πιο καταξιωμένα στον κόσμο του Ελεύθερου Λογισμικού.

Βιβλιογραφία

*Η ιστορία του Unix BSD είναι ενδεικτική ενός ιδιόμορφου τρόπου ανάπτυξης λογισμικού κατά τη δεκαετία '70 και '80. Όποιος ενδιαφέρεται περισσότερο γι' αυτό, μπορεί να απολαύσει την ανάγνωση του βιβλίου "Twenty years of Berkeley Unix" (Marshall Kirk McKusick, 1999) [170], το οποίο παρακολουθεί την εξέλιξη, από την ταϊνία που ο Bob Fabry έφερε στο Berkeley, με την ιδέα να έφτιαχνε μία από τις πρώτες εκδόσεις της συνάρτησης κώδικα (code function) κατά Thompson και Ritchie, σε ένα PDP-11 (που είχε αγοραστεί από κοινού από τις Σχολές Πληροφορικής, Στατιστικής και Μαθηματικών), μέχρι τις νομικές διαμάχες που ξεκίνησε η AT&T και ως τις πλέον πρόσφατες εκδόσεις του κώδικα που οδήγησαν στην γέννηση της οικογένειας *BSD των ελεύθερων λειτουργικών συστημάτων.*

2.2.3. Οι απαρχές του Διαδικτύου

Σχεδόν από την δημιουργία του, κατά τη δεκαετία του '70, το Διαδίκτυο συσχετίστηκε στενά με το Ελεύθερο Λογισμικό. Από τη μια μεριά, από το ξεκίνημα, η κοινότητα των developers που έχτισε το Διαδίκτυο, είχε πολλές ξεκάθαρες αρχές, που θα γινόταν αργότερα κλασσικές στον κόσμο του Ελεύθερου Λογισμικού. Για παράδειγμα, η σημασία των χρηστών που είναι σε θέση να βοηθούν στον εντοπισμό και διόρθωση των bugs ή που ανταλλάσσουν κώδικα. Η σημασία του BSD Unix στην ανάπτυξή του (παρέχοντας, κατά τη δεκαετία του '80, την πλέον δημοφιλή υλοποίηση των πρωτοκόλλων TCP/IP), κατέστησε εύκολη τη μεταφορά πολλών συνηθειών και τρόπων εργασίας

από την κοινότητα- οι developers επικεντρώθηκαν γύρω από το CSRG- σε μία άλλη κοινότητα – την κοινότητα των developers που έφτιαχναν εκείνο που λεγόταν, τότε, NSFNet και που θα γινόταν αργότερα το Διαδίκτυο- και τούμπαλιν. Πολλές από τις βασικές εφαρμογές για την ανάπτυξη του Διαδικτύου, όπως το *Sendmail* (διακομιστής αλληλογραφίας) ή το *BIND* (υλοποίηση υπηρεσιών ονόματος χώρου/ domain name) ήταν ελεύθερες και, σε μεγάλο βαθμό, αποτελούσαν την έκβαση της συνεργασίας μεταξύ αυτών των δύο κοινοτήτων.

Τελικά, κατά τα τέλη της δεκαετίας του '80 και στην δεκαετία του '90, η κοινότητα του Ελεύθερου Λογισμικού ήταν ένας από τους πρώτους που εξερεύνησε εις βάθος τις νέες δυνατότητες που προσέφερε το Διαδίκτυο, προς τις διάφορες γεωγραφικά διάσπαρτες ομάδες, για να μπορέσουν να συνεργαστούν. Σε μεγάλο βαθμό, αυτή η εξερεύνηση κατέστησε εφικτή ακόμη και την ίδια την ύπαρξη της κοινότητας BSD, το FSF ή την ανάπτυξη του GNU/Linux.

Μία από τις πλέον ενδιαφέρουσες πλευρές της ανάπτυξης του Διαδικτύου, από την σκοπιά του Ελεύθερου Λογισμικού, ήταν η εντελώς ελεύθερη διαχείριση των εγγράφων του και των κανόνων του. Αν και σήμερα μπορεί να μοιάζει φυσιολογικό (διότι είναι σύνηθες, π.χ. στην IETF ή στο World Wide Web Consortium), εκείνο τον καιρό, η ελεύθερη διάθεση όλων των προδιαγραφών του και των εγγράφων σχεδίασης, περιλαμβανομένων και των κανόνων που ορίζουν τα πρωτόκολλα, ήταν κάτι το επαναστατικό και κεφαλαιώδους σημασίας για την ανάπτυξή του. Στο *Netizens*, σχετικά με την ιστορία και την επίδραση του *Usenet* και του Διαδικτύου [139] (σελ. 106), μπορούμε να διαβάσουμε:

"Αυτή η ανοιχτή διαδικασία παρότρυνε και οδήγησε στην ανταλλαγή πληροφορίας. Η τεχνολογική ανάπτυξη είναι εφικτή τότε μόνο όταν η πληροφορία επιτρέπεται να κυκλοφορεί ελεύθερα και εύκολα μεταξύ των ενδιαφερομένων μερών. Η ενθάρρυνση της συμμετοχής είναι η βασική αρχή που έκανε την ανάπτυξη του Διαδικτύου εφικτή."

Μπορούμε να αντιληφθούμε γιατί ο κάθε developer που είναι αναμεμιγμένος σε κάποιο project Ελεύθερου Λογισμικού, θα παρείχε σχεδόν αμέριστη υποστήριξη στην παραπάνω παράγραφο.

Σε ένα άλλο σημείο από το *"The evolution of packet switching"* [195] (σελ. 267) μπορούμε να διαβάσουμε:

"Αφού το ARPANET ήταν ένα δημόσιο project που συνέδεε πολλά μεγάλα Πανεπιστήμια και Ερευνητικά ιδρύματα, οι λεπτομέρειες της υλοποίησης και της απόδοσης δημοσιεύθηκαν ευρέως."

Προφανώς, αυτό συνήθως συμβαίνει με τα projects Ελεύθερου Λογισμικού, όπου όλη η πληροφορία σχετική με ένα project (και όχι μόνο με την υλοποίηση), καθίσταται συνήθως δημόσια.

Μέσα σε αυτή την ατμόσφαιρα και στην προ Διαδικτύου εποχή, στα μισά της δεκαετίας του '90, δημιουργήθηκε ένα ολόκληρο business, η σχέση ανάμεσα στην κοινότητα των χρηστών και τους developers απέκτησε κρίσιμη σημασία. Ήταν ακριβώς κατά τη διάρκεια εκείνης της περιόδου που πολλές οργανώσεις έμαθαν να μην εμπιστεύονται έναν μοναδικό προμηθευτή υπηρεσιών επικοινωνίας και δεδομένων, αλλά περισσότερο έναν σύνθετο συνδυασμό εταιρειών παροχής υπηρεσιών, κατασκευής εξοπλισμού, επαγγελματιών developers και εθελοντών, κλπ. Οι καλύτερες υλοποιήσεις πολλών προγραμμάτων δεν ήταν εκείνες που ερχόταν "πακέτο" μαζί με το λειτουργικό σύστημα και αγοράζοταν μαζί με το hardware, αλλά μάλλον οι ελεύθερες υλοποιήσεις, που τις αντικαθιστούσαν πολύ γρήγορα. Οι πιο επιτυχημένες αναπτύξεις δεν ήταν έκβαση των ερευνητικών σχεδίων κάποιας μεγάλης εταιρείας, αλλά περισσότερο το προϊόν φοιτητών ή επαγγελματιών, οι οποίοι έθεσαν υπό δοκιμή κάποιες ιδέες και συνέλεξαν ανάδραση (*feedback*) από τους διάφορους χρήστες αυτών των ελεύθερων προγραμμάτων.

Όπως έχουμε ήδη αναφέρει, το Διαδίκτυο προσέφερε, επίσης, στο Ελεύθερο Λογισμικό τα βασικά εργαλεία για την συνεργασία εξ αποστάσεως. Η ηλεκτρονική αλληλογραφία, τα news groups, οι υπηρεσίες ανώνυμου FTP (που υπήρξαν τα πρώτα μαζικά αποθετήρια Ελευθέρου Λογισμικού) και, αργότερα, τα *web-based* ολοκληρωμένα συστήματα ανάπτυξης, ήταν βασικής σημασίας (και απαραίτητα) για τη ανάπτυξη της κοινότητας του Ελευθέρου Λογισμικού, όπως την γνωρίζουμε σήμερα, και ιδιαιτέρως, για τη λειτουργία της συντριπτικής πλειοψηφίας των projects Ελευθέρου Λογισμικού. Από την αρχή, κάποια projects σαν το GNU ή το BSD, έκανα μαζική και εντατική χρήση όλων αυτών των μηχανισμών, αναπτύσσοντας συγχρόνως, καθώς τα χρησιμοποιούσαν, και νέα εργαλεία, που με τη σειρά τους βελτιώναν το Διαδίκτυο.

Βιβλιογραφία

Οι αναγνώστες που ενδιαφέρονται για την ανάπτυξη του Διαδικτύου, έτσι όπως γράφτηκε από τους περισσότερους βασικούς του πρωταγωνιστές, μπορούν να συμβουλευθούν το "*A brief history of the Internet*" (εκδόθηκε από την ACM, 1997) [166].

2.2.4. Άλλα projects

Κατά τη δεκαετία του 1980, γεννήθηκαν πολλά άλλα σημαντικά projects Ελευθέρου Λογισμικού. Υπογραμμίζουμε για τη σημασία τους αλλά και για τη μελλοντική τους βαρύτητα, τον X Window (σύστημα διαχείρισης παραθύρων για Unix-οειδή συστήματα), που αναπτύχθηκε στο MIT, ένα από τα πρώτα παραδείγματα μίας χρηματοδότησης ευρείας κλίμακος για ένα Ελεύθερο project, που να υποστηρίχθηκε από μία κοινοπραξία επιχειρήσεων. Αξίζει επίσης να μνημονεύσουμε το Ghostscript, ένα σύστημα διαχείρισης εγγράφων *PostScript*, που το ανέπτυξε μία εταιρεία ονόματι Aladdin Software, που υπήρξε μία από τις πρώτες περιπτώσεις αναζήτησης ενός επιχειρηματικού μοντέλου βασισμένου στην παραγωγή Ελευθέρου Λογισμικού.

Κατά τα τέλη της δεκαετίας του 1980, είχε ήδη δημιουργηθεί ένας ολόκληρος αστερισμός μικρών (αλλά και όχι-και-τόσο-μικρών) projects ελεύθερου Λογισμικού που τρέχανε. Όλα αυτά, μαζί με τα μεγαλύτερα projects που αναφέραμε ως τώρα, έθεσαν τις βάσεις για τα πρώτα απολύτως ελεύθερα συστήματα, τα οποία εμφανίσθηκαν στις αρχές της δεκαετίας του 1990.

2.3. Όλα παίρνουν το δρόμο τους

Γύρω στο 1990, τα περισσότερα συστατικά μέρη ενός πλήρους συστήματος ήταν έτοιμα υπό την μορφή του Ελευθέρου Λογισμικού. Από τη μια μεριά, το GNU project και οι διανομές BSD είχαν ολοκληρώσει τις περισσότερες εφαρμογές που απαρτίζουν ένα λειτουργικό σύστημα. Από την άλλη μεριά, διάφορα projects, σαν το X Window ή το ίδιο το GNU, είχαν ήδη φτιάξει από παραθυρικά περιβάλλοντα μέχρι και μεταγλωττιστές (*compilers*), που ήταν, συχνά, ανάμεσα στα καλύτερα του είδους τους (π.χ., πολλοί διαχειριστές της SunOS ή συστημάτων Ultrix, αντικαθιστούσαν διάφορες εφαρμογές των *proprietary* εφαρμογών των συστημάτων τους, με τις ελεύθερες εκδόσεις του GNU ή του BSD, για τους χρήστες τους). Για να ολοκληρωθεί ένα πλήρες σύστημα, χτισμένο αποκλειστικά με Ελεύθερο Λογισμικό, έλειπε μόνον ένα συστατικό: ο πυρήνας. Δύο χωριστές και ανεξάρτητες προσπάθειες ήρθαν να συμπληρώσουν το κενό: το 386BSD και το Linux.

2.3.1. Η αναζήτηση για έναν πυρήνα

Κατά τα τέλη της δεκαετίας του '80 και στις αρχές της δεκαετίας του '90, το GNU project διέθετε μία βασική σειρά από utilities και εργαλεία, που καθιστούσαν εφικτή την επίτευξη ενός πλήρους λειτουργικού συστήματος. Ακόμη και τότε, πολλές ελεύθερες εφαρμογές, περιλαμβανομένης της ιδιαίτερης περίπτωσης του X Window, ανήκαν ανάμεσα στις καλύτερες της κατηγορίας τους (*Unix utilities*, μεταγλωττιστές...). Εν πάσει περιπτώσει, για να συμπληρωθεί το puzzle, έλειπε ένα βασικό κομμάτι: ο πυρήνας του λειτουργικού συστήματος. Το GNU project, με ένα project ονόματι Hurd, αναζητούσε αυτό το τμήμα που έλλειπε, με την πρόθεση να χτίσει έναν πυρήνα χρησιμοποιώντας σύγχρονες τεχνολογίες.

2.3.2. Η οικογένεια *BSD

Ουσιαστικά, κατά την ίδια ακριβώς περίοδο, η κοινότητα BSD ήταν, επίσης, σε αναζήτηση ενός ελεύθερου πυρήνα. Από τη διανομή Net-2, έλειπαν μόνο έξι αρχεία, για την ολοκλήρωσή του (τα υπόλοιπα είχαν φτιαχτεί από το CSRG ή τους συνεργάτες του). Στις αρχές του 1992, ο Bill Jolitz ολοκλήρωσε αυτά τα αρχεία και διένειμε το 386BSD, ένα σύστημα που λειτούργησε στην αρχιτεκτονική i386 και που, συν τω χρόνω, θα οδηγούσε στη δημιουργία των projects NetBSD, FreeBSD και OpenBSD. Η πρόοδος κατά τους επόμενους μήνες ήταν γρήγορη και, μέχρι το τέλος του έτους, ήταν ήδη αρκετά σταθερό για να μπορεί να χρησιμοποιηθεί σε περιβάλλοντα “μη-κρίσιμης” παραγωγής, που συμπεριελάμβαναν, για παράδειγμα, ένα παραθυρικό περιβάλλον, χάρη στο XFree project (που είχε οδηγήσει στη γέννηση του X Window, για την αρχιτεκτονική i386), ή έναν πολύ ποιοτικό μεταγλωττιστή, τον GCC. Αν και υπήρχαν συστατικά που χρησιμοποιούσαν άλλες άδειες (όπως εκείνες από τα GNU projects, που έκαναν χρήση της GPL), το μεγαλύτερο μέρος του συστήματος είχε διανεμηθεί υπό την άδεια BSD.

Βιβλιογραφία

Μερικά επεισόδια, από αυτή την περίοδο, αναδεικνύουν τις δυνατότητες των μοντέλων ανάπτυξης του Ελεύθερου Λογισμικού. Υπάρχει η πασίγνωστη περίπτωση του Linus Torvalds, που ανέπτυξε τον πυρήνα Linux, ενόσω ήταν δευτεροετής φοιτητής στο Πανεπιστήμιο του Helsinki. Αλλά αυτή δεν είναι η μόνη περίπτωση ενός φοιτητή που βρίσκει τον δρόμο του και προοδεύει, χάρη στις δικές του ελεύθερες εφαρμογές. Για παράδειγμα, ο Γερμανός Thomas Roel έκανε προσαρμογή (*port*) το X11R4 (μία έκδοση του *X Window system*) σε ένα PC που βασιζόταν σε ένα 386. Αυτή η ανάπτυξη τον οδήγησε να εργασθεί για την Dell, και αργότερα να γίνει ο ιδρυτής του X386 και του XFree project, που ήταν ουσιώδη για να προσδώσουν γρήγορα ένα παραθυρικό περιβάλλον στο GNU/Linux και στα λειτουργικά *BSD. Μπορείτε να διαβάσετε περισσότερα σχετικά με την ιστορία του XFree και τον ρόλο του Roel σε αυτό, στο άρθρο “The history of xFree86” (*Περιοδικό Linux*, Δεκέμβριος 1991) [135].

Έπειτα, ήρθε η αγωγή εκ μέρους της USL, που έκανε πολλούς εν δυνάμει χρήστες να φοβούνται τις μηνύσεις εναντίον τους, αν το Πανεπιστήμιο της Καλιφόρνιας έχανε την δικαστική υπόθεση, είτε απλώς, αν το όλο project βάλτωνε. Ίσως, αυτός ήταν ο λόγος που αργότερα, η εγκατεστημένη βάση του GNU/Linux ήταν πολύ μεγαλύτερη από όλα τα λειτουργικά *BSD, υπολογιζόμενα αθροιστικά. Αλλά δεν είμαστε σε θέση να το γνωρίζουμε με σιγουριά αυτό.

2.3.3. Το GNU/Linux ανεβαίνει στο προσκήνιο

Τον Ιούλιο του 1991, ο Linus Torvalds (ένας Φινλανδός φοιτητής 21 ετών) έστειλε το πρώτο του μήνυμα, όπου αναφερόταν στο σχέδιό του (κατά την περίοδο εκείνη) να κτίσει ένα Ελεύθερο σύστημα παρόμοιο με το Minix. Τον Σεπτέμβριο δημοσίευσε την αρχική έκδοση (0.01), και από τότε, κάθε λίγες εβδομάδες, εμφανιζόταν και κάποιες νέες εκδόσεις. Τον Μάρτιο του 1994, εμφανίστηκε η έκδοση 1.0, η πρώτη που ονομάστηκε και “σταθερή”, αν και ο αρχικός πυρήνας που είχε φτιάξει ο Linus είχε ήδη χρησιμοποιηθεί για πολλούς μήνες. Κατά την περίοδο αυτή, στην

κυριολεξία, εκατοντάδες προγραμματιστών/developers εστράφησαν στο Linux, ενσωματώνοντας όλα τα λογισμικά του έργου GNU, καθώς και το XFree και πολλά παραπάνω Ελεύθερα προγράμματα. Διαφορετικά από την προσέγγιση των *BSD, ο πυρήνας του Linux και ένα μεγάλο μέρος των στοιχείων που ενσωματώθηκαν σε αυτό, κυκλοφορούσαν υπό την άδεια GPL.

Βιβλιογραφία

Η ιστορία για το Linux είναι, πιθανώς, μία από τις πλέον ενδιαφέρουσες (και πιο γνωστές) στον κόσμο του Ελεύθερου Λογισμικού. Μπορείτε να βρείτε πολλούς δεσμούς (*links*) που παραπέμπουν σε πληροφορίες σχετικές με αυτό, από τις σελίδες που γιορτάζουν την 10η επέτειο από την αναγγελία του, αν και κατά πάσα πιθανότητα, ένα από τα πιο ενδιαφέροντα είναι το "History of Linux", του Ragib Hasan [138]. Από περιέργεια, μπορείτε επίσης να ανατρέξετε στο νήμα στο οποίο ο Linus Torvalds ανακοίνωνε εκείνο που, αργότερα, θα γινότανε το Linux (στο *newsgroup comp.os.minix*) στη διεύθυνση <http://groups.google.com/groups?th=d161e94858c4c0b9>. Εκεί εξηγεί πως εργαζόταν πάνω στον πυρήνα του από τον Απρίλιο και πως είχε ήδη ενσωματώσει σε αυτόν (*ported*) μερικά εργαλεία του GNU project (ανέφερε συγκεκριμένα το *Bash* και τον *GCC*).

Ανάμεσα στις διάφορες ιδέες που αναπτύχθηκαν γύρω από το σύστημα GNU/Linux, μία από τις πιο ενδιαφέρουσες ήταν το concept της *Διανομής*⁽¹⁾. Σύντομα εμφανίσθηκαν οι πρώτες διανομές, κατά το 1992 (MCC Interim Linux, του Πανεπιστημίου του Manchester, του TAMU, στο Texas της εταιρείας A&M, και η πιο διάσημη, η SLS, που αργότερα οδήγησε στο Slackware, το οποίο και κυκλοφορούσε ακόμη και την πρώτη δεκαετία του 2000), σκιαγραφώντας την άφιξη της ανταγωνιστικότητας, μέσα στο πλαίσιο των συστημάτων που πακετάρονταν με επίκεντρο το GNU/Linux. Η κάθε διανομή προσπαθεί να προσφέρει ένα έτοιμο προς χρήση σύστημα GNU/Linux, και αρχίζοντας από την ίδια κοινή βάση Λογισμικού, προσπαθεί να ανταγωνιστεί μέσω της επινόησης βελτιώσεων που θεωρούνται σημαντικές από τη βάση των δικών της χρηστών. Πέρα από το να προσφέρει προμεταγλωττισμένα και έτοιμα προς χρήση πακέτα, οι διάφορες διανομές τείνουν, επίσης, να προσφέρουν και τα δικά τους εργαλεία για την όλη διαχείριση, εγκατάσταση, αντικατάσταση και διαγραφή αυτών των πακέτων, πέραν της αρχικής εγκατάστασης στον υπολογιστή, καθώς επίσης και τη διαχείριση και επίβλεψη του Λειτουργικού Συστήματος.

⁽¹⁾ Αυτή η έννοια επεξηγείται με λεπτομέρεια στο αντίστοιχο λήμμα της Wikipedia, www.wikipedia.org/wiki/Linux_distribution

Συν τω χρόνω, οι διανομές διαδεχόταν η μία την άλλη, καθώς, κατά καιρούς, επικρατούσε σε δημοτικότητα κάποια από αυτές. Από όλες αυτές, θα αναφέρουμε τις παρακάτω:

- 1) Η Διανομή Debian, που αναπτύχθηκε από μία κοινότητα εθελοντών χρηστών.
- 2) Η Red Hat Linux, που στην αρχή αναπτύχθηκε ενδοεταιρικά, από την εταιρεία Red Hat, αλλά η οποία, αργότερα, υιοθέτησε ένα μοντέλο βασισμένο περισσότερο στην κοινότητα, που οδήγησε στο Fedora Core.
- 3) Η Διανομή Suse, που οδήγησε στο OpenSUSE, ακολουθώντας μία παρόμοια εξέλιξη με εκείνη της Red Hat.
- 4) Η Διανομή Mandriva (διάδοχος της *Mandrake Linux* και της *Conectiva*).
- 5) Η Διανομή Ubuntu, που προήλθε από το Debian και παρήχθη με βάση αυτό, από την εταιρεία Canonical.

2.4. Καιρός για ωριμότητα

Κάπου στα μισά της δεκαετίας του 2000, το ίδιο το GNU/Linux, αλλά και το OpenOffice.org ή το Firefox, ήταν αρκετά συχνά παρόντα στις συζητήσεις των ΜΜΕ. Στη συντριπτική τους πλειοψηφία, οι εταιρείες χρησιμοποιούν Ελεύθερο Λογισμικό, για τουλάχιστον ένα τμήμα των διαφόρων διαδικασιών τους στο χώρο του IT. Είναι, επίσης, δύσκολο να είναι κανείς φοιτητής στον τομέα IT και να μην χρησιμοποιεί ευρύτατα Ελεύθερο Λογισμικό. Δηλαδή, το Ελεύθερο Λογισμικό δεν αποτελεί πλέον μία απλή υποσημείωση στον τομέα της Πληροφορικής (IT) και έχει ήδη καταστεί κάτι το πολύ σημαντικό στον τομέα αυτό. Τώρα όμως αρχίζει να καθίσταται στρατηγικής σημασίας και για εταιρείες IT, εταιρείες του δευτερογενούς τομέως (εκείνες που χρησιμοποιούν εντατικά το Λογισμικό, ακόμη κι αν η πρωτογενής τους δραστηριότητα είναι διαφορετική) καθώς και για Δημόσιους Οργανισμούς.

Και αργά αλλά σταθερά, διεισδύει και στο χώρο των οικιακών χρηστών. Από αυτή τη γενικότερη άποψη, εισερχόμαστε σε μία περίοδο ωρίμανσης. Εν κατακλείδι όλων των παραπάνω, άρχισε να προκύπτει ένα σημαντικό ερώτημα, το οποίο εμπεριέχει, κατά κάποιο τρόπο, όλα αυτά που συμβαίνουν: "μήπως είμαστε αντιμέτωποι με ένα νέο μοντέλο βιομηχανίας λογισμικού;" Μπορεί, ενδεχομένως, το Ελεύθερο Λογισμικό να μην εξελιχθεί τελικά σε τίποτε παραπάνω από μία παροδική μόδα, την οποία θα θυμόμαστε με νοσταλγία μία ημέρα. Αλλά μπορεί επίσης (και μοιάζει να τείνει όλο και περισσότερο προς τα εκεί), να αποδειχθεί ότι πρόκειται όντως για ένα νέο μοντέλο, που ήρθε εδώ, για να μείνει, και ίσως, για να αλλάξει δραστικά μία από τις πιο πρόσφατες μορφές βιομηχανίας των καιρών μας, αλλά και μία από εκείνες με την μεγαλύτερη επιρροή.

2.4.1. Τα τέλη της δεκαετίας του 1990

Περί τα μέσα της δεκαετίας, το Ελεύθερο Λογισμικό μπορούσε ήδη να προσφέρει ολοκληρωμένα περιβάλλοντα (τις διανομές του συστήματος GNU/Linux, τα συστήματα *BSD...) που μπορούσαν να υποστηρίξουν την καθημερινή δουλειά των περισσότερων ανθρώπων, και ειδικότερα των προγραμματιστών και developers λογισμικού. Υπήρχαν, βεβαίως, ακόμη αρκετές εκκρεμότητες (μεταξύ των οποίων η κυριότερη ήταν να αναπτυχθεί μία καλύτερη γραφική διεπαφή για τους χρήστες, σε μία περίοδο όπου τα Windows 95 αποτελούσαν το σημείο αναφοράς), αλλά υπήρχαν, ήδη, αρκετές χιλιάδες ανθρώπων, σε παγκόσμια κλίμακα, που χρησιμοποιούσαν αποκλειστικά Ελεύθερο Λογισμικό για την εκτέλεση των καθημερινών τους εργασιών. Ανακοινώνονταν διάφορα νέα projects σε γρήγορη εναλλαγή και, έτσι, το Ελεύθερο Λογισμικό ξεκίνησε το μακρύ του ταξίδι προς τον εταιρικό κόσμο, τα media και την ευρύτερη συνειδητοποίηση του κοινού γενικότερα.

Αυτή ακριβώς η περίοδος συνδέθηκε και με την απογείωση του Διαδικτύου, ως ενός μέσου για την δικτύωση όλων, που σε πλείστες περιπτώσεις καθοδηγήθηκε χέρι-χέρι από τα Ελεύθερα Λογισμικά (ειδικά σε ό,τι αφορά την υποδομή του). Η άφιξη του Διαδικτύου στα σπίτια εκατομμυρίων τελικών χρηστών, εδραίωσε περαιτέρω αυτή την κατάσταση, τουλάχιστον στον χώρο των διακομιστών: οι πλέον διάσημοι web (HTTP) servers ήταν ανέκαθεν Ελεύθεροι (πρώτα ο διακομιστής NCSA και, στη συνέχεια, ο Apache).

Ίσως, την αρχή του δρόμου που χάραξε το Ελεύθερο Λογισμικό, μέχρι την πλήρη του διάδοση στο ευρύ κοινό, να την περιγράφει με τον καλύτερο τρόπο το διάσημο δοκίμιο του Eric Raymond, *"The cathedral and the bazaar"* (*"Ο Καθεδρικός ναός και το παζάρι"*, Eric S. Raymond, 2001) [192]. Παρ' ότι τα περιεχόμενα του βιβλίου ήταν, ήδη, γνωστά πράγματα στην κοινότητα των προγραμματιστών/ developers Ελεύθερου Λογισμικού, το γεγονός το ίδιο πως γράφτηκαν σε ένα κομμάτι χαρτί και διανεμήθηκαν σε ευρεία κλίμακα, το κατέστησαν ένα εργαλείο με μεγάλη επιρροή στην προώθηση

της ιδέας του *Ελεύθερου Λογισμικού*, ως ενός εναλλακτικού μηχανισμού παραγωγής και διανομής, που θα μπορούσε να χρησιμοποιηθεί, σε αντίθεση με τους έως τότε συνηθισμένους τρόπους της κλασικής βιομηχανίας λογισμικού. Ένα άλλο σημαντικό κείμενο εκείνης της περιόδου ήταν το *"Setting up shop. The Business of open source software"* [141], του Frank Hecker, που για πρώτη φορά περιέγραφε τα νέα εν δυνάμει επιχειρηματικά μοντέλα για το Ελεύθερο Λογισμικό, και το οποίο γράφτηκε για να επηρεάσει την απόφαση δημοσίευσης του κώδικα του Netscape Navigator.

Ενώ το κείμενο του Raymond ήταν ένα σπουδαίο εργαλείο για την προώθηση μερικών από τα βασικότερα χαρακτηριστικά του Ελεύθερου Λογισμικού, η δημοσίευση του κώδικα του Netscape Navigator ήταν η πρώτη περίπτωση μίας σχετικά μεγάλης εταιρείας, σε ένα πολύ καινοτόμο τομέα (την μόλις τότε ανατέλλουσα βιομηχανία του Διαδικτύου), που πήρε την απόφαση να κυκλοφορήσει ένα από τα προϊόντα της ως Ελεύθερο Λογισμικό. Εκείνη την εποχή, ο Netscape Navigator έχανε τον πόλεμο των περιηγητών έναντι του προϊόντος της Microsoft (του Internet Explorer) και αυτό, εν μέρει, χάρη στην τακτική της Microsoft να το ενσωματώνει στο λειτουργικό της σύστημα. Πολλοί θεωρούν ότι ο Netscape έκανε το μόνο πράγμα που μπορούσε να πράξει: δηλαδή να δοκιμάσει την αλλαγή των κανόνων, για να καταφέρει να ανταγωνιστεί με έναν γίγαντα. Και από αυτήν ακριβώς την αλλαγή των κανόνων (προσπαθώντας να ανταγωνισθεί με ένα μοντέλο Ελεύθερου Λογισμικού), γεννήθηκε το Mozilla project. Αυτό το έργο, που είχε τα δικά του προβλήματα, οδήγησε αρκετά χρόνια αργότερα, σε έναν Περιηγητή (*navigator*) που, παρότι δεν ανέκτησε την τεράστια φέτα αγοράς που είχε κατακτήσει ο Netscape την περίοδο εκείνη, εν τούτοις, από τεχνολογικής πλευράς, εξελίχθηκε σε ένα καλό προϊόν, τουλάχιστον τόσο, όσο και οι ιδιοταγείς ανταγωνιστές του.

Σε κάθε περίπτωση, ανεξάρτητα από τις μετέπειτα επιτυχίες του, η ανακοίνωση του Netscape ότι θα δημοσίευε τον πηγαίο κώδικα του Περιηγητή, είχε μία βαθιά επίδραση στην βιομηχανία λογισμικού. Πολλές βιομηχανίες λογισμικού άρχισαν τότε να λαμβάνουν υπ' όψη τους το Ελεύθερο Λογισμικό ως κάτι που άξιζε την προσοχή τους.

Αλλά και οι χρηματαγορές άρχισαν να εστιάζουν την προσοχή τους στο Ελεύθερο Λογισμικό. Μέσα στο κλίμα ευφορίας που δημιουργήθηκε με τη φούσκα των εταιρειών dot.com, πολλές εταιρείες Ελεύθερου Λογισμικού κατέστησαν στόχοι διάφορων επενδυτών. Ίσως, η πιο διάσημη περίπτωση είναι εκείνη της Red Hat, μία από τις πρώτες εταιρείες που συνειδητοποίησαν ότι η πώληση CD, με έτοιμα προς χρήση συστήματα GNU/Linux, μπορούσε, εν δυνάμει, να εξελιχθεί σε ένα επιχειρηματικό μοντέλο. Η Red Hat άρχισε να διανέμει τη δική της έκδοχή ενός Red Hat Linux, θέτοντας τεράστια έμφαση, τουλάχιστον σε σχέση με τα κοινώς ισχύοντα τον καιρό εκείνο, στην ευκολία χρήσης του συστήματος, καθώς και στην ευκολία συντήρησης από ανθρώπους που δεν διέθεταν ιδιαίτερες τεχνικές γνώσεις Πληροφορικής. Με το πέρασμα του χρόνου, διαφοροποιήθηκε, αν και παραμένοντας πάντα μέσα στη σφαίρα του Ελεύθερου Λογισμικού και, κατά τον Σεπτέμβριο του 1998, ανακοίνωσε ότι η Intel και η Netscape είχαν επενδύσει επάνω της. "Αν είναι καλό για την Intel και την Netscape, τότε πρέπει να είναι καλό και για μας", είναι η σκέψη που πρέπει να έκαναν πολλοί επενδυτές τότε. Όταν η Red Hat μπήκε στο Χρηματιστήριο το καλοκαίρι του 1999, η είσοδός της σημαδεύθηκε από μία υπερκάλυψη των αιτούμενων κεφαλαίων και, συντόμως, η αξία της μετοχής της εκτοξεύθηκε θεαματικά. Ήταν η πρώτη φορά που μία εταιρεία αντλούσε χρηματοδότηση από το Χρηματιστήριο Αξιών, με ένα μοντέλο που βασιζόταν στο Ελεύθερο Λογισμικό. Αλλά δεν ήταν και η μόνη: αργότερα, και άλλες εταιρείες, σαν την VA Linux ή την Andover.net (η οποία και εξαγοράσθηκε, στη συνέχεια, από την VA Linux), έπραξαν το ίδιο.

Σημείωση

Η Red Hat παρέχει έναν κατάλογο, με την επίτευξη των διάφορων σταδίων ανάπτυξης της εταιρείας στην διεύθυνση <http://fedora.redhat.com/about/history/>.

Κατά την περίοδο αυτή, προέκυψαν και πολλές άλλες εταιρείες, που βασιζόταν στο επιχειρηματικό μοντέλο του Ελεύθερου Λογισμικού. Παρότι δεν εισήχθησαν στο Χρηματιστήριο, ούτε και πέτυχαν τέτοιες τρομερές εμπορικές υπεραξίες, υπήρξαν, εν τούτοις, πολύ σημαντικές για τη συνολική ανάπτυξη του Ελεύθερου Λογισμικού. Εμφανίσθηκαν, για παράδειγμα, πολλές εταιρείες που άρχισαν να διανέμουν τις δικές τους εκδόσεις του συστήματος GNU/Linux, σαν τη SuSE (Γερμανία), την Conectiva (Βραζιλία) ή την Mandrake (Γαλλία), που ενώθηκε στη συνέχεια με την προηγούμενη, για να σχηματίσουν την Mandriva. Άλλες, πάλι, προσέφεραν υπηρεσίες σε εταιρείες που χρειαζόταν συντήρηση ή προσπαθούσαν να προσαρμόσουν ελεύθερα προϊόντα: η LinuxCare (US), η Alcove (Γαλλία), η ID Pro (Γερμανία), και πολλές άλλες.

Στο εν τω μεταξύ, οι γίγαντες του χώρου άρχισαν να παίρνουν θέση σχετικά με το Ελεύθερο Λογισμικό. Μερικές εταιρείες, σαν την IBM, το ενσωμάτωσαν κατ' ευθείαν μέσα στη στρατηγική τους. Άλλες, σαν τη Sun Microsystems, ανέπτυξαν μία περίεργη σχέση με αυτό, κατά καιρούς υποστηρίζοντάς το, σε άλλες περιόδους παραμένοντας αδιάφορη, και σε άλλες περιστάσεις επιδεικνύοντας εχθρική συμπεριφορά. Οι περισσότερες (όπως οι Apple, Oracle, HP, SGI, κλπ.) διερεύνησαν το μοντέλο του Ελεύθερου Λογισμικού με διάφορες στρατηγικές, που ποίκιλλαν από την επιλεκτική απελευθέρωση του Λογισμικού, μέχρι και την πλήρη προσαρμογή (*porting*) των προϊόντων τους στο σύστημα GNU/Linux. Κάπου ανάμεσα σε αυτά τα δύο άκρα, υπήρχαν και πολλά άλλα ενδιάμεσα σχέδια δράσης, όπως η εκτεταμένη (σε μικρότερο ή μεγαλύτερο βαθμό) και εντατική χρήση του Ελεύθερου Λογισμικού στα προϊόντα τους (η περίπτωση του Mac OS X), είτε η διερεύνηση επιχειρηματικών μοντέλων που βασίζονται στην παροχή συντήρησης των προϊόντων Ελεύθερου Λογισμικού.

Από τεχνικής απόψεως, το πιο αξιοσημείωτο συμβάν αυτής της περιόδου ήταν, κατά πάσα πιθανότητα, η εμφάνιση δύο φιλόδοξων projects, που σχεδιάσθηκαν για τη μεταφορά του Ελεύθερου Λογισμικού στο *περιβάλλον της επιφάνειας εργασίας* των μη έμπειρων χρηστών IT: το KDE και το GNOME. Για να το πούμε απλά, ο τελικός σκοπός ήταν να μπορεί να επικοινωνεί ο χρήστης με το GNU/Linux, ή με τα συστήματα *BSD και με όλα τα προγράμματά τους, χωρίς να χρειάζεται να χρησιμοποιήσει τη γραμμή εντολών.

Το έργο KDE ξεκίνησε δημόσια τον Οκτώβριο του 1996. Χρησιμοποιώντας τις βιβλιοθήκες γραφικών Qt (που εκείνον τον καιρό ήταν ένα ιδιοταγές προϊόν, που ανήκε στην εταιρεία Trolltech, αλλά χωρίς χρέωση για τη χρήση του σε περιβάλλον GNU/Linux²), άρχισε η δημιουργία μίας σειράς εφαρμογών για την επιφάνεια εργασίας, οι οποίες μπορούσαν να δουλέψουν με έναν ολοκληρωμένο τρόπο και με κοινή αισθητική και εμφάνιση. Τον Ιούλιο 1998, δόθηκε στη δημοσιότητα η έκδοση 1.0 του περιβάλλοντος εργασίας του K Desktop και την οποία ακολούθησε, σύντομα, μία σειρά ωριμότερων εκδόσεων, με ολοένα αυξανόμενο βαθμό ωριμότητας. Οι διάφορες διανομές GNU/Linux ενσωμάτωσαν σύντομα το KDE, ως επιφάνεια εργασίας για τους χρήστες τους (ή, τουλάχιστον, σαν μία επιλογή για το περιβάλλον επιφάνειας εργασίας που θα μπορούσαν να επιλέξουν οι χρήστες).

⁽²⁾ Αργότερα, η Qt άρχισε να διανέμεται υπό την ελεύθερη άδεια QPL (*Qt Public Licence*), που είναι ασύμβατη με την GPL και η οποία δημιούργησε αρκετά προβλήματα, αφού το μεγαλύτερο κομμάτι του KDE διενέμετο υπό την GPL. Εν ευθέτω χρόνω, η Trolltech αποφάσισε, επιτέλους, να διανείμει την Qt υπό την άδεια GPL, θέτοντας έτσι ένα τέρμα σε αυτά τα προβλήματα.

Κυρίως λόγω αντίδρασης στην εξάρτηση του KDE από τις ιδιοταγείς βιβλιοθήκες της Qt, τον Αύγουστο του 1997, δόθηκε στη δημοσιότητα το GNOME project (*Miguel de Icaza, "Η ιστορία του GNOME Project"*) [101], με παρόμοια χαρακτηριστικά και σκοπούς με εκείνα του KDE, αλλά με τη ρητή δέσμευση και το στόχο να συμπεριλαμβάνει στα συστατικά του μόνον Ελεύθερο

Λογισμικό. Τον Μάρτιο του 1999, κυκλοφόρησε η έκδοση 1.0 του GNOME, η οποία επρόκειτο και αυτή να βελτιωθεί και να γίνει πιο σταθερή, με το πέρασμα του χρόνου. Από εκείνη τη στιγμή και μετά, οι περισσότερες διανομές Ελεύθερων Λειτουργικών συστημάτων (καθώς και πολλών από τα διάφορα Unix-οειδή ιδιοταγή αντίστοιχα), προσέφεραν σαν επιλογή είτε το GNOME, είτε το KDE, για το περιβάλλον της επιφάνειας εργασίας και τις εφαρμογές για αμφότερα τα περιβάλλοντα.

Στο εντωμεταξύ, τα κυριότερα projects Ελεύθερου Λογισμικού που έτρεχαν, έχαιραν άκρας υγείας, και νέα projects προέκυπταν, σχεδόν, καθημερινά. Σε διάφορες επί μέρους, εξειδικευμένες αγορές (*niche markets*), το Ελεύθερο Λογισμικό κρίθηκε ως η καλύτερη λύση (με διεθνή σχεδόν αναγνώριση). Για παράδειγμα, από την εμφάνισή του κατά τον Απρίλιο του 1995, ο Apache διατήρησε το μεγαλύτερο μερίδιο αγοράς των διακομιστών Διαδικτύου (*web servers*). Το XFree86, το Ελεύθερο project που ανέπτυξε τον διαχειριστή παραθύρων X Window, αποτελεί, με διαφορά, την πιο δημοφιλή έκδοχή του X Window (και, άρα, και το πιο εκτεταμένο σύστημα διαχείρισης παραθύρων για Unix-οειδή συστήματα). Ο GCC αναγνωρίστηκε ως ο πιο φορητός μεταγλωττιστής C και από εκείνους με την καλύτερη ποιότητα. Το GNAT, το σύστημα μεταγλώττισης για την Ada95, κατέκτησε το μεγαλύτερο μερίδιο αγοράς των μεταγλωττιστών Ada, μέσα σε μόλις λίγα χρόνια, και ούτω καθ' εξής.

Το 1998, ιδρύθηκε η Πρωτοβουλία για το Ανοικτό Λογισμικό (Open Source Initiative, OSI), και αποφάσισε να υιοθετήσει τον όρο “Λογισμικό Ανοικτού Κώδικα”, σαν ένα εμπορικό σήμα (brand), για να μπορέσει το Ελεύθερο Λογισμικό να διεισδύσει στον επιχειρηματικό κόσμο, αποφεύγοντας, συγχρόνως, την αμφισημία του (Αγγλικού) όρου “free” (ο οποίος μπορεί να σημαίνει τόσο “Ελεύθερο” προς χρήση, αλλά και “χωρίς χρέωση”, “Δωρεάν”). Αυτή η συζήτηση πυροδότησε μία από τις εντονότερες αντιπαραθέσεις στον κόσμο του Ελεύθερου Λογισμικού (που συνεχίζει έως και σήμερα), αφού το Ίδρυμα Ελεύθερου Λογισμικού (*Free Software Foundation*), αλλά και άλλοι, θεωρούσαν ότι ήταν πολύ ορθότερο να μιλάμε για *Ελεύθερο Λογισμικό* (Richard Stallman, “Why free software is better than open source”, 1998) [206].

Σε κάθε περίπτωση, το OSI ξεκίνησε μία έντονη εκστρατεία προώθησης του νέου εμπορικού του σήματος (brand), το οποίο και υιοθετήθηκε από πολλούς ως ο προτιμότερος τρόπος για να μιλάμε περί Ελεύθερου Λογισμικού και, ειδικότερα, στον Αγγλόφωνο κόσμο. Για να ορίσει το *Λογισμικό Ανοικτού Κώδικα*, το OSI χρησιμοποίησε έναν ορισμό που προήλθε από εκείνον που είχε το Debian project για να ορίσει “τι εστί” Ελεύθερο Λογισμικό. (“*Debian free software guidelines*”, http://www.debian.org/social_contract.html#guidelines) [104], που συγχρόνως, αντανάκλούσε αρκετά στενά την ιδέα που το FSF είχε για το θέμα αυτό (“*Free software definition*”, <http://www.gnu.org/philosophy/free-sw.html>) [120], πράγμα που σημαίνει ότι, από πρακτικής απόψεως, σχεδόν οποιοδήποτε πρόγραμμα που θεωρείται *Ελεύθερο Λογισμικό*, μπορεί επίσης να θεωρείται και *Ανοικτού Κώδικα* και το αντίθετο. Εν πάσει περιπτώσει, οι Κοινότητες του Ελεύθερου Λογισμικού και του Λογισμικού Ανοικτού Κώδικα (ή τουλάχιστον, οι άνθρωποι που τις αντιπροσωπεύουν) μπορεί να είναι πολύ διαφορετικές.

2.4.2. Δεκαετία του 2000

Στα πρώτα χρόνια της δεκαετίας του 2000, το Ελεύθερο Λογισμικό ήταν, ήδη, ένας σοβαρός ανταγωνιστής στον χώρο των διακομιστών και άρχισε να είναι έτοιμο και για το desktop.

Τα συστήματα σαν το GNOME, το KDE, το OpenOffice.org και το Mozilla Firefox, μπορούν να χρησιμοποιηθούν από οικιακούς χρήστες και επαρκούν για να καλύψουν τις ανάγκες πολλών εταιρειών, τουλάχιστον σε ότι αφορά τις εφαρμογές γραφείου. Τα Ελεύθερα Συστήματα (και ειδικά τα συστήματα που βασίζονται στο GNU/ Linux) είναι εύκολα στην εγκατάσταση, και η συνθετότητα της συντήρησης και της ενημέρωσής τους είναι συγκρίσιμη με εκείνες που απαιτούνται για τα ιδιοταγή συστήματα.

Προς το παρόν, κάθε εταιρεία της βιομηχανίας λογισμικού, διαθέτει μία στρατηγική για το Ελεύθερο Λογισμικό. Οι περισσότερες από τις κορυφαίες πολυεθνικές (IBM, HP, Sun, Novell, Apple, Oracle...) ενσωματώνουν το Ελεύθερο Λογισμικό σε μεγαλύτερο ή μικρότερο βαθμό. Στη μία άκρη, μπορούμε να βρούμε εταιρείες σαν την Oracle, που αντιδρούν, απλούστατα, με την προσαρμογή (*porting*) των προϊόντων τους σε GNU/Linux. Από την άλλη άκρη, μπορούμε να βρούμε την IBM, που επιδεικνύει την πλέον δραστήρια στρατηγική και λάνσαρε τις μεγαλύτερες εκστρατείες διαφήμισης σχετικά με το GNU/Linux. Ανάμεσα στους ηγέτες της αγοράς IT, μόνον η Microsoft τοποθετήθηκε ξεκάθαρα σε αντίθεση με το Ελεύθερο Λογισμικό και, ιδιαίτερα, με το Λογισμικό που διανέμεται υπό την άδεια GPL.

Όσο αφορά τον κόσμο του Ελεύθερου Λογισμικού αυτόν καθ' εαυτόν, παρά και τις αντιπαραθέσεις που περιστασιακά ταράζουν την Κοινότητα, παρατηρείται μία μαζική ανάπτυξη. Υπάρχουν όλο και περισσότεροι developers, κάθε ημέρα, καθώς και περισσότερα δραστήρια projects λογισμικού, περισσότεροι χρήστες, κλπ. Κάθε μέρα που περνά, το Ελεύθερο Λογισμικό απομακρύνεται από το περιθώριο και καθίσταται μία υπολογίσιμη δύναμη, με την οποία πρέπει κανείς να αναμετρηθεί.

Από τη σκοπιά αυτή, αναδύονται νέα πεδία του επιστητού (*disciplines*), που μελετούν ιδιαίτερα τον τομέα του Ελεύθερου Λογισμικού, και πράγματα όπως η Τεχνολογία του Ελεύθερου Λογισμικού (*Free Software engineering*). Βασιζόμενοι στην έρευνα, αρχίζουμε λίγο-λίγο να κατανοούμε πως λειτουργεί το Ελεύθερο Λογισμικό στις διάφορες πτυχές του: μοντέλα ανάπτυξης, επιχειρηματικά μοντέλα, μηχανισμοί συντονισμού των developers, διαχείριση έργων Ελεύθερου Λογισμικού, κίνητρα των developers, κλπ.

Αυτά τα χρόνια, αρχίζουμε, επίσης, να βλέπουμε τα πρώτα αποτελέσματα της “υπερακτιοποίησης” (*off-shoring*), που το Ελεύθερο Λογισμικό καθιστά εφικτό: χώρες που εθεωρούντο “περιφερειακές”, τώρα συμμετέχουν ενεργά στον κόσμο του Ελεύθερου Λογισμικού. Για παράδειγμα, ο αριθμός των Μεξικανών ή Ισπανών developers (αμφότερες, χώρες με περιορισμένη παράδοση στη βιομηχανία λογισμικού), σε projects σαν το GNOME, είναι σημαντική (*Lancashire, "Code, culture and cash: the fading altruism of open source development", 2001*) [164]. Ο ρόλος της Βραζιλίας είναι ακόμη πιο ενδιαφέρων, με τους πολυάριθμους developers και ειδικούς που διαθέτει στις τεχνολογίες Ελεύθερου Λογισμικού, καθώς και μία αποφασιστική υποστήριξη από την Δημόσια Διοίκηση. Η περίπτωση του gnuLinEx αξίζει ιδιαίτερη προσοχή, ως ένα παράδειγμα για το πως μία περιοχή με πολύ μικρή παράδοση στην ανάπτυξη λογισμικού, μπορεί να προσπαθήσει να αλλάξει την κατάσταση, μέσω μίας επιθετικής στρατηγικής υιοθέτησης και ενσωμάτωσης Ελεύθερου Λογισμικού.

Από τη σκοπιά της λήψης αποφάσεων, όταν συζητάμε για υλοποίηση λύσεων Ελεύθερου Λογισμικού, πρέπει να τονίσουμε το γεγονός πως υπάρχουν μερικοί τομείς της αγοράς (όπως οι υπηρεσίες Διαδικτύου ή οι Εφαρμογές Γραφείου), για τους οποίους το Ελεύθερο Λογισμικό αποτελεί μία φυσική επιλογή, που δεν μπορεί να αγνοήσει κανείς, όταν εξετάζει ποιο σύστημα να χρησιμοποιήσει.

Από την αρνητική πλευρά, μέσα στα χρόνια αυτά, αποδείχθηκε πως το νομικό περιβάλλον εντός του οποίου δρα το Ελεύθερο Λογισμικό αλλάζει, διεθνώς, με μεγάλη ταχύτητα. Από τη μια μεριά, τα διπλώματα ευρεσιτεχνίας λογισμικού (*software patents*) υιοθετούνται όλο και περισσότερο, από ολόένα και περισσότερες χώρες. Από την άλλη μεριά, νέοι νόμοι περί κατοχύρωσης πνευματικών δικαιωμάτων (*copyright*), καθιστούν δύσκολη ή και αδύνατη την ανάπτυξη Ελεύθερων εφαρμογών σε κάποιους τομείς, ανάμεσα στους οποίους είναι διάσημοι οι DVD viewers (λόγω των αλγόριθμων κωδικοποίησης CSS που χρησιμοποιεί αυτή η τεχνολογία).

To gnuLinEx

Στις αρχές του 2002, η Κυβέρνηση της Περιφέρειας της Extremadura ανακοίνωσε δημόσια το έργο gnuLinEx. Η ιδέα ήταν απλή: να προωθηθεί η δημιουργία μίας Διανομής που να βασίζεται στο GNU/Linux, με κεντρικό σκοπό να χρησιμοποιηθεί σε χιλιάδες υπολογιστών, που επρόκειτο να εγκατασταθούν στα Δημόσια Σχολεία όλης της Περιφέρειας Extremadura, η οποία βρίσκεται στη Δυτική Ισπανία, στα σύνορα με την Πορτογαλία, έχει περίπου 1.000.000 κατοίκους και δεν διακρίθηκε ποτέ για τις τεχνολογικές της πρωτοβουλίες. Πράγματι, η περιοχή αυτή δεν διέθετε, πρακτικά, καμία βιομηχανία λογισμικού.

Σε αυτό το πλαίσιο, το gnuLinEx προσέφερε μία πολύ ενδιαφέρουσα συνεισφορά στο πανόραμα του Ελεύθερου Λογισμικού, σε παγκόσμια κλίμακα. Πέραν από το να είναι απλώς μία νέα διανομή GNU/Linux, βασισμένη στο Debian (πράγμα που παραμένει ακόμη αξιοσημείωτα ασυνήθιστο), και ανεξαρτήτως της τεράστιας επιρροής στα MME (ήταν η πρώτη φορά που η περιοχή Extremadura απετέλεσε αντικείμενο της πρώτης σελίδας της εφημερίδας *The Washington Post*, αλλά και η πρώτη φορά για ένα προϊόν Ελεύθερου Λογισμικού), εκείνο που ήταν εξαιρετικό (τουλάχιστον φαινομενικά), ήταν η στιβαρή υποστήριξη μίας Δημόσιας Διοίκησης προς το Ελεύθερο Λογισμικό. Η Διοίκηση της Περιφέρειας της Extremadura αποφάσισε να δοκιμάσει ένα διαφορετικό μοντέλο, σχετικά με το Εκπαιδευτικό Λογισμικό και, στη συνέχεια, εφήρμοσε αυτό το μοντέλο σε όλα τα Λογισμικά που θα χρησιμοποιούσε στο πλαίσιο της δικής της επικράτειας. Έτσι, έγινε η πρώτη Δημόσια Διοίκηση μίας ανεπτυγμένης χώρας που υιοθέτησε αποφασιστικά αυτή την προσέγγιση.

Δημιουργήθηκε μεγάλο ενδιαφέρον γύρω από αυτή την πρωτοβουλία της Διοίκησης της Περιφέρειας, τόσο στην ίδια την Extremadura, όσο και εκτός αυτής: υπάρχουν Πανεπιστημιακές Σχολές που διδάσκουν IT χρησιμοποιώντας το gnuLinEx, έχουν γραφτεί βιβλία για την υποστήριξη αυτής της διδασκαλίας, πωλούνται υπολογιστές με το gnuLinEx ως προεγκατεστημένο Λειτουργικό. Σε γενικές γραμμές, προσπαθούν να φτιάξουν έναν ολόκληρο εκπαιδευτικό και επιχειρηματικό ιστό γύρω από αυτή την εμπειρία, για να μπορέσει να στηριχθεί το όλο εγχείρημα. Έγινε, επίσης, εξαγωγή αυτής της εμπειρίας. Στις αρχές του 21ου αιώνα, διάφορες αυτόνομες κοινότητες στην Ισπανία, υποστήριζαν τη χρήση Ελεύθερου Λογισμικού στην Εκπαίδευση (με τον έναν ή τον άλλο τρόπο), και γενικότερα, αναγνωρίστηκε ευρύτερα η σημασία του για την Δημόσια Διοίκηση.

To Knoppix

Από τα τέλη της δεκαετίας του 1990, υπήρχαν διανομές του GNU/Linux που μπορούσαν να εγκατασταθούν με ευκολία, αλλά το Knoppix, του οποίου η πρώτη έκδοση εμφανίστηκε το 2002, ήταν εκείνη που, πιθανώς, επέτρεψε σε αυτή την ιδέα να αγγίξει την πλήρη της έκφραση. Πρόκειται για ένα CD που είναι εκκινήσιμο σχεδόν από τον κάθε υπολογιστή, μετατρέποντάς τον (χωρίς καν να απαιτείται “φορμάρισμα” του σκληρού δίσκου, αφού μπορεί να τρέξει “live”), σε ένα πλήρως λειτουργικό μηχάνημα GNU/Linux, με μία επιλογή των συνηθέστερων εργαλείων. Το Knoppix συνδυάζει την καλή αυτόματη αναγνώριση υλικού/hardware, με μία καλή επιλογή προγραμμάτων και με τη δυνατότητα να “τρέχει” live. Για παράδειγμα, επιτρέπει να πάρει κανείς μία γρήγορη και άμεση εμπειρία για το τι σημαίνει να δουλεύει σε περιβάλλον GNU/Linux. Οδήγησε, επίσης, στη δημιουργία μίας ολόκληρης οικογένειας Διανομών του ίδιου τύπου, με ειδικεύση στις ιδιαίτερες απαιτήσεις ενός προφίλ χρήστη.

To OpenOffice.org

Το 1999, η εταιρεία Sun Microsystems εξαγόρασε μία άλλη Γερμανική εταιρεία, ονόματι

StarDivision, της οποίας το πιο διάσημο προϊόν ήταν το StarOffice, μία σουίτα εφαρμογών γραφείου, παρόμοια σε λειτουργίες με το σύνολο εργαλείων που διέθετε το Microsoft Office. Ένα έτος αργότερα, η Sun δημοσίευσε το μεγαλύτερο μέρος του κώδικα του StarOffice υπό την ελεύθερη Άδεια GPL, δημιουργώντας έτσι το έργο OpenOffice.org. Αυτό το project κυκλοφόρησε την πρώτη έκδοση 1.0 του OpenOffice.org, τον Μάιο του 2002. Το OpenOffice.org εξελίχθηκε σε μία ποιοτική σουίτα εφαρμογών γραφείου, που προσέφερε παρόμοια λειτουργικότητα με εκείνη που είχε οποιοδήποτε άλλο προϊόν γραφείου, και, το πιο σημαντικό, είχε συμβατή και πολύ καλή διαλειτουργικότητα με τις μορφές αρχείων και τα δεδομένα του Microsoft Office. Αυτά τα χαρακτηριστικά το καθιέρωσαν σαν μία εφαρμογή αναφοράς του Ελεύθερου Λογισμικού, για τον κόσμο των ενοποιημένων εφαρμογών (*suites*) γραφείου.

Η σημασία του OpenOffice.org, από την άποψη της επέκτασης της χρήσης του Ελεύθερου Λογισμικού, σε έναν ευρύτατο αριθμό χρηστών, είναι τεράστια. Κατέστη, επιτέλους, εφικτή η μετάβαση, σχεδόν χωρίς προβλήματα, από τα ιδιοταγή κλειστά περιβάλλοντα για τις συνηθισμένες σουίτες γραφείου (αναμφίβολα, τις διασημότερες εφαρμογές επιχειρηματικού Λογισμικού στον κόσμο), σε πλήρως ελεύθερα περιβάλλοντα (σαν το GNU/Linux συν το GNOME και/ή το KDE, συν το OpenOffice.org). Πέραν αυτού, η μετάβαση μπορεί να γίνει πολύ ομαλά: από τη στιγμή που το OpenOffice.org δουλεύει καλά και σε περιβάλλον Windows της Microsoft, δεν είναι απαραίτητο να αλλάξει κανείς Λειτουργικό Σύστημα, για να μπορέσει να πειραματισθεί σε βάθος με τη χρήση του Ελεύθερου Λογισμικού.

To Mozilla, ο Firefox και οι υπόλοιποι

Από πρακτικής απόψεως, από την εμφάνισή του, από το 1994 μέχρι το 1996, ο Netscape Navigator ήταν ο αναντίρρητος ηγέτης της αγοράς των web browsers, με μερίδια αγοράς μέχρι και το 80%. Η κατάσταση άρχισε να αλλάζει όταν η Microsoft συμπεριέλαβε τον Internet Explorer μέσα στα Windows 95, οδηγώντας τον Netscape Navigator στη σταδιακή απώλεια μεριδίων αγοράς. Στις αρχές του 1998, ο Netscape ανακοίνωσε ότι θα διένειμε ένα μεγάλο μέρος του κώδικα του προγράμματος πλοήγησης ως Ελεύθερο Λογισμικό, το οποίο και έπραξε το Μάρτιο του ιδίου έτους, ξεκινώντας το Mozilla project. Για αρκετό χρονικό διάστημα, το project υπέφερε από αβεβαιότητα, ή ακόμη και από απαισιοδοξία (για παράδειγμα, όταν ο ηγέτης του, ο Jamie Zawinski, το εγκατέλειψε), καθώς αν και ο καιρός περνούσε από την ίδρυση του έργου, δεν προέκυπτε ένα από προϊόν. Τον Ιανουάριο του 2000, το project παρήγαγε το Mozilla M13, που θεωρήθηκε και η πρώτη σχετικά σταθερή έκδοση. Τον Μάιο του 2002, δημοσιεύθηκε, επιτέλους, η έκδοση 1.0, η πρώτη επισήμως σταθερή έκδοση, πάνω από τέσσερα χρόνια μετά την πρώτη απελευθέρωση του κώδικα του Netscape Navigator.

Βιβλιογραφία

Στο "Netscape Navigator", του Brian Wilson, [234], μπορούμε να συμβουλευθούμε έναν λεπτομερή κατάλογο των βασικών εκδόσεων του Netscape Navigator και του Mozilla, καθώς και των βασικών τους χαρακτηριστικών.

Επιτέλους, ο Mozilla έγινε πραγματικότητα, αν και, ίσως, πολύ καθυστερημένα, αν λάβουμε υπ' όψη τα μερίδια αγοράς που, στο εντωμεταξύ, είχε κατακτήσει ο Internet Explorer, κατά το 2002 ή το 2003 (όταν ήταν ο αναμφισβήτητος ηγέτης, αφήνοντας τον Mozilla και τους άλλους σε μία εντελώς περιθωριακή θέση). Παρά το ότι πήρε τόσο πολύ χρόνο, το Mozilla project απέφερε καρπούς: όχι μόνο τον φυσιολογικώς αναμενόμενο καρπό (τον περιηγητή διαδικτύου Mozilla), αλλά και άλλους "παράπλευρους" καρπούς: για παράδειγμα τον Firefox, έναν άλλον περιηγητή που βασιζόταν στην ίδια μηχανή HTML, που μετετράπη σε κυρίως προϊόν και ο οποίος, από τότε που πρωτοεμφανίσθηκε το 2005, καταφέρνει να διαβρώσει σιγά-σιγά το μερίδιο αγοράς των άλλων προγραμμάτων περιήγησης του διαδικτύου.

Το Mozilla project βοήθησε στην κάλυψη ενός μεγάλου κενού στον κόσμο του Ελεύθερου Λογισμικού. Πριν την εμφάνιση του Konqueror (του περιηγητή που ανήκε στο KDE project), δεν υπήρχαν πολλοί ελεύθεροι περιηγητές που διέθεταν μία γραφική διεπαφή. Από την κυκλοφορία του Mozilla και μετά, προέκυψε ένας τεράστιος αριθμός projects που βασιζόταν σε αυτό, και τα οποία κατέληξαν στην παραγωγή ενός αντίστοιχα μεγάλου αριθμού περιηγητών. Στο ίδιο χρονικό διάστημα, ο συνδυασμός του Mozilla Firefox και του OpenOffice.org, επέτρεψε στο Ελεύθερο Λογισμικό να μπορεί να χρησιμοποιείται για τις πιο συνηθισμένες δουλειές, ακόμη και μέσα στο περιβάλλον Windows της Microsoft (και τα δύο αυτά προγράμματα δουλεύουν όχι μόνο σε περιβάλλον GNU/Linux, *BSD και άλλων Unix-οειδών συστημάτων, αλλά και σε περιβάλλον Windows). Για πρώτη φορά στην ιστορία του Ελεύθερου Λογισμικού, η μετάβαση από το κλειστό ιδιοταγές στο Ελεύθερο Λογισμικό, για περιβάλλοντα γραφείου, κατέστη μία απλή υπόθεση: μπορούμε να αρχίσουμε να χρησιμοποιούμε αυτές τις δύο εφαρμογές σε περιβάλλον Windows, χωρίς καμία αλλαγή Λειτουργικού Συστήματος (για όσους το χρησιμοποιούν ως το κανονικό τους Λ.Σ.), και σταδιακά με τον καιρό, να καταργήσουμε τα μόνα εναπομείναντα Μη-Ελεύθερα τμήματα και να μεταβούμε σε περιβάλλον GNU/Linux ή FreeBSD.

Η περίπτωση του SCO

Στις αρχές του 2003, η SCO corporation (γνωστή, προηγουμένως, ως Caldera Systems & Caldera International) υπέβαλλε μήνυση κατά της IBM, υποστηρίζοντας ότι η τελευταία παραβίασε τα πνευματικά της δικαιώματα. Αν και η περίπτωση αυτή ήταν σύνθετη, εστιασθηκε στην υποστήριξη της άποψης ότι η IBM συνεισέφερε στον πυρήνα Linux με Κώδικα που ανήκε στην SCO. Το Μάιο του 2007, η νομική αυτή διαμάχη δεν είχε ακόμη επιλυθεί και είχε γίνει ακόμη πιο περίπλοκη από περαιτέρω μηνύσεις (της IBM και της Red Hat κατά της SCO, της SCO κατά της AutoZone και της DaimlerChrysler, δύο μεγάλων χρηστών IT) καθώς και από τις εκστρατείες της SCO που απειλούσε τις μεγάλες εταιρείες που χρησιμοποιούσαν το Linux, κλπ.

Παρά το ότι ο νικητής αυτής της γιγαντιαίας νομικής διαμάχης δεν έχει ακόμη ανακηρυχθεί, η όλη περίπτωση ανέδειξε ορισμένες πλευρές που αφορούν το Ελεύθερο Λογισμικό. Πιο συγκεκριμένα, πολλές εταιρείες έλαβαν υπ' όψη τα προβλήματα που ενδεχομένως να αντιμετώπιζαν, αν χρησιμοποιήσουν το Linux και άλλα ελεύθερα προγράμματα, και την εγγύηση ότι πράττοντας έτσι, δεν παραβιάζουν τα πνευματικά ή τα επιχειρηματικά δικαιώματα τρίτων.

Κατά κάποιο τρόπο, αυτή η περίπτωση, όπως και άλλες (σαν εκείνες που αφορούσαν την εγκυρότητα των αδειών GPL, που επιλύθηκαν στην Γερμανία το 2005), μπορούν να ερμηνευθούν και σαν ένα σημάδι ωριμότητας του Ελεύθερου Λογισμικού. Έπαυσε να είναι ένας ξένος μέσα στον επιχειρηματικό κόσμο και έγινε τμήμα πολλών από τις δραστηριότητές του (περιλαμβανομένων εκείνων που σχετίζονται με νομικές στρατηγικές).

Το Ubuntu, η Canonical, η Fedora και η Red Hat

Παρά το ότι η Canonical (η εταιρεία που παράγει και διανέμει το Ubuntu) θα μπορούσε να θεωρηθεί σαν ένας νεοεισερχόμενος στον κύκλο των διανομών GNU/Linux, οι δραστηριότητές της αξίζουν την προσοχή μας. Σε ένα σχετικά μικρό χρονικό διάστημα, το Ubuntu καθιερώθηκε σαν μία από τις περισσότερο γνωστές και τις πιο ευρέως χρησιμοποιούμενες διανομές, που χαίρει φήμης για καλή ποιότητα και για μεγάλη ευκολία εγκατάστασης και χρήσης. Το Ubuntu ξεχωρίζει, επίσης, για τη μεγαλύτερη προσοχή που δίνει στο να συμπεριλαμβάνει κατά βάση Ελεύθερο Λογισμικό, σε σχέση με τις περισσότερες άλλες διανομές που αναπτύσσονται από εταιρείες.

Σε κάθε περίπτωση, τα βασικά χαρακτηριστικά του Ubuntu (και της στρατηγικής της Canonical) ήταν να στηριχθούν πάνω στο Debian, μία διανομή που φτιάχθηκε και συντηρείται από εθελοντές. Στην πραγματικότητα, το Ubuntu δεν είναι η πρώτη περίπτωση διανομής που βασίζεται στο Debian

(μία άλλη πασίγνωστη περίπτωση είναι το gnuLinEx), αλλά είναι, ίσως, αυτή που έγινε αποδέκτης της μεγαλύτερης χρηματοδότησης. Για παράδειγμα, η Canonical μίσθωσε πολυάριθμους ειδικούς της Debian (πολλοί από τους οποίους συμμετέχουν στο project) και ακολούθησε μία στρατηγική που αναζητά τη συνεργασία με το Εθελοντικό project. Σε κάποιο βαθμό, η Canonical προσπάθησε να συμπληρώσει αυτό που θεωρούσε πως έλειπε από το Debian, για να κερδίσει την αποδοχή του μέσου χρήστη.

Η Red Hat, με την σειρά της, ακολούθησε μία άλλη οδό για να καταλήξει σε μία αρκετά παρόμοια κατάσταση. Αρχίζοντας από μία διανομή που ανέπτυξε με εξ ολοκλήρου δικούς της πόρους, αποφάσισε να συνεργασθεί με την Fedora, μία ομάδα εθελοντών που ήδη εργαζόταν σε διανομές βασισμένες στην Red Hat, ούτως ώστε να παραγάγει το Fedora Core, δηλαδή τη "λαϊκή" διανομή. Η Red Hat διατηρεί και την δική της εκδοχή για εταιρείες, αλλά αυτή η συνεργασία με τους εθελοντές, είναι, εν τέλει, πολύ παρόμοια με εκείνη που υλοποίησε το Ubuntu.

Ίσως, όλα αυτά τα κινήματα δεν είναι άλλο παρά το αποτέλεσμα ισχυρού ανταγωνισμού που παρατηρείται στην αγορά των διανομών GNU/Linux και ακόμη μία αξιοσημείωτη τάση: της συνεργασίας μεταξύ εταιρειών και εθελοντών των κοινοτήτων για παραγωγή Ελεύθερου Λογισμικού.

Παραμετροποιημένες (Customised) διανομές

Αφότου το Linux ανέβηκε στο προσκήνιο, ένας μεγάλος αριθμός ομάδων και εταιρειών δημιούργησαν τις δικές τους διανομές, που βασίζονται σε αυτό. Αλλά σε όλα αυτά τα χρόνια, το φαινόμενο αυτό "έπιασε τόπο" σε πολλές οργανώσεις και εταιρείες που ήθελαν να έχουν τις δικές τους παραμετροποιημένες εκδόσεις, που να καλύπτουν τις δικές τους απαιτήσεις. Η παραμετροποίηση κατάφερε να επεκταθεί διότι η διαδικασία έγινε φθηνότερη και υπάρχει άφθονη διαθεσιμότητα τεχνικής γνώσης για να επιτευχθεί αυτό, ακόμη και μετατρέποντάς την σε μία εξειδικευμένη αγορά (*niche market*) για κάποιες εταιρείες.

Ίσως μία από τις καλύτερα γνωστές περιπτώσεις παραμετροποιημένων διανομών είναι εκείνη των αυτόνομων κοινοτήτων της Ισπανίας. Η Διοίκηση της Περιφέρειας της Extremadura, με το gnuLinEx που ανέπτυξε, πυροδότησε μία τάση που, έκτοτε, ακολούθησαν πολλές άλλες αυτόνομες κοινότητες. Η διαδικασία αυτή έγινε τόσο συνηθισμένη, ώστε πολλές από αυτές προκηρύσσουν τακτικά διαγωνισμούς για την δημιουργία και την συντήρηση νέων εκδόσεων των διανομών τους. Η ανάπτυξη παραμετροποιημένων διανομών υλοποιεί μία τάση που ο κόσμος του Ελεύθερου Λογισμικού ήδη συζητούσε επί μακρόν: την προσαρμογή των προγραμμάτων σε συγκεκριμένες ανάγκες χρηστών, χωρίς να χρειάζεται οι αρχικοί δημιουργοί να είναι απαραίτητως εκείνοι που υλοποιούν την προσαρμογή.

Βιβλιογραφία

Μερικές από τις πιο γνωστές διανομές του GNU/Linux στις αυτόνομες Ισπανικές κοινότητες περιλαμβάνουν:

- το gnuLinEx: <http://linex.org> (Extremadura)
- το Guadalinex: <http://guadalinex.org> (Andalucía)
- το Lliurex: <http://lliurex.net> (Comunidad Valenciana)
- το Augustux: <http://www.zaralinux.org/proy/augustux/> (Aragón)
- το MAX: http://www.educa.madrid.org/web/madrid_linux/ (Madrid)
- το MoLinux: <http://molinux.info> (Castilla-La Mancha)

Συνεργασίες Εταιρείας-με-εταιρεία και Εθελοντών-με-Εταιρεία

Ουσιαστικά, από τις απαρχές του Ελεύθερου Λογισμικού, υπήρξαν εταιρείες που συνεργάστηκαν

με εθελοντές, στην ανάπτυξη εφαρμογών. Εν πάσει περιπτώσει, σε αυτά τα χρόνια, στη φάση που μοιάζει να φθάνουμε στην ωριμότητα, υπάρχει ένας αυξανόμενος αριθμός εταιρειών που χρησιμοποιούν Ελεύθερο Λογισμικό ως τμήμα της στρατηγικής τους για συνεργασία με άλλες εταιρείες, όταν το βρίσκουν ενδιαφέρον. Δύο από τις πιο αξιόλογες περιπτώσεις, που οργανώθηκαν ειδικά γι' αυτό τον σκοπό, είναι το ObjectWeb (μία συμμαχία που σχηματίστηκε στη Γαλλία, αλλά που με τον καιρό, εξελίχθηκε καθαρά σε διεθνή) και το Morfeo (στην Ισπανία). Σε αμφότερες τις περιπτώσεις, μία ομάδα εταιρειών συμφώνησαν να αναπτύξουν ένα σύνολο ελεύθερων συστημάτων που τους ενδιαφέρουν, και αποφάσισαν να τα διανείμουν ως Ελεύθερο Λογισμικό.

Σε άλλες περιπτώσεις, διάφορες εταιρείες προσπάθησαν δραστήρια να συνεργασθούν για την ανάπτυξη Έργων Ελεύθερου Λογισμικού που προωθούνται από εθελοντές, είτε προσπάθησαν να πείσουν τους εθελοντές να συνεργασθούν με τα δικά τους ελεύθερα projects. Το Ίδρυμα GNOME ή το ήδη προαναφερθέν Ubuntu, σε σχέση με το Debian, αποτελούν παραδείγματα αυτού του πρώτου σεναρίου. Η Sun με το OpenOffice.org και το OpenSolaris, είτε η Red Hat με το Fedora Core, αποτελούν παραδείγματα για το δεύτερο.

Επέκταση σε άλλους τομείς

Το Ελεύθερο Λογισμικό απέδειξε ότι στο χώρο της ανάπτυξης προγραμμάτων, υπάρχει ένας άλλος τρόπος για να γίνονται τα πράγματα. Στην πράξη, είδαμε πως η παραχώρηση της ελευθερίας για διανομή, τροποποίηση και χρήση, μπορούν να πετύχουν τη βιωσιμότητα, είτε μέσω εθελοντικής δουλειάς, είτε μέσω δημιουργίας επιχειρήσεων που επιτρέπουν στις εταιρείες να επιβιώνουν.

Καθώς ο χρόνος περνά, αυτή η ίδια ιδέα μεταφέρεται σε άλλους τομείς των πνευματικών έργων. Οι Άδειες Creative Commons κατέστησαν εφικτή την απελευθέρωση ολόκληρων τομέων, όπως της λογοτεχνίας, της μουσικής, ή του video. Η Wikipedia αποδεικνύει ότι ακόμη και ένας χώρος τόσο ιδιαίτερος, σαν εκείνον της παραγωγής εγκυκλοπαιδειών, μπορεί να κινηθεί σε πολύ ενδιαφέρουσες πορείες. Και υπάρχουν όλο και περισσότεροι λογοτέχνες συγγραφείς, μουσικά συγκροτήματα, ακόμη και παραγωγοί κινηματογραφικών ταινιών, που ενδιαφέρονται για άλλα μοντέλα ελεύθερης παραγωγής και διανομής.

Σε όλους αυτούς τους τομείς, υπάρχει ακόμη πολύ δουλειά να γίνει και, σχεδόν σε όλους αυτούς, η πράξη δεν έχει ακόμη πλήρως αποδείξει ότι είναι εφικτή μία βιώσιμη δημιουργία με τα ελεύθερα μοντέλα. Αλλά δεν μπορούμε και να αρνηθούμε ότι ο πειραματισμός με αυτά, προσεγγίζει το “σημείο βρασμού”.

Το Ελεύθερο Λογισμικό ως αντικείμενο μελέτης

Αν και ορισμένα έργα, σαν το διάσημο *"The cathedral and the bazaar"*, έστρωσαν τον δρόμο για την μελέτη του Ελεύθερου Λογισμικού αυτού καθ' εαυτού, χρειάστηκε ως και το 2001 και τα μετέπειτα χρόνια, για να αρχίσει η Ακαδημαϊκή Κοινότητα να λαμβάνει υπ' όψη της το Ελεύθερο Λογισμικό ως κάτι άξιο μελέτης. Συν τω χρόνω, η μαζική διαθεσιμότητα στοιχείων (σχεδόν τα πάντα στον κόσμο του Ελεύθερου Λογισμικού είναι δημόσια και διαθέσιμα, μέσα από δημόσια αρχεία πληροφόρησης) και οι καινοτομίες που προσέφερε το Ελεύθερο Λογισμικό, προσήλκυσαν την προσοχή πολλών ομάδων. Κάπου στα μισά της δεκαετίας του 2000, σημειώθηκαν ήδη πολλαπλά διεθνή συνέδρια που επικεντρωνόταν ειδικά στο Ελεύθερο Λογισμικό, διάφορα κορυφαία περιοδικά δημοσίευαν συχνά άρθρα γύρω από αυτό, και διάφοροι Οργανισμοί χρηματοδότησης ερευνών άνοιξαν τις στρόφιγγες, στοχεύοντας ειδικά σε αυτό.

2.5. Το μέλλον: ένας αγώνας δρόμου μετ' εμποδίων;

- Ασφαλώς, είναι δύσκολο να προβλέψουμε το μέλλον, ούτε και είναι ο σκοπός μας. Άρα, αντί να προσπαθούμε να προβλέψουμε πως θα μοιάζει το μέλλον του Ελεύθερου Λογισμικού, θα προσπαθήσουμε να φωτίσουμε τα προβλήματα με τα οποία συχνά θα έρθει αντιμέτωπο και που, πραγματικά, ήδη αντιμετωπίζει εδώ και καιρό. Το πως ο κόσμος του Ελεύθερου Λογισμικού θα καταφέρει να ανταπεξέλθει σε αυτά τα εμπόδια, θα καθορίσει αναμφίβολα και την όλη του κατάσταση, μετά από πολλά χρόνια.
- Φ.ΑΒ.ΑΜ. (Φόβος, Αβεβαιότητα, Αμφιβολία/ *FUD, Fear, Uncertainty, Doubt*). Αυτή είναι μία τεχνική που χρησιμοποιείται αρκετά συχνά στον κόσμο των Τεχνολογιών της Πληροφορίας (IT), και χρησιμοποιείται από τους ανταγωνιστές του Ελεύθερου Λογισμικού για να δυσφημίσουν το Ελεύθερο Λογισμικό, με περισσότερες ή λιγότερες δικαιολογίες και ποικίλους βαθμούς επιτυχίας. Σε γενικές γραμμές, το Ελεύθερο Λογισμικό παρέμεινε αρκετά άνοσο προς αυτού του είδους τις τεχνικές, ίσως, χάρη στην πολυπλοκότητά του και στους διαφορετικούς τρόπους μέσω των οποίων διεισδύει στον εταιρικό κόσμο.
- Διάλυση (*Dissolution*). Πολλές εταιρείες δοκιμάζουν τα όρια του Ελεύθερου Λογισμικού ως μοντέλου, και προσπαθούν, ειδικά, να προσφέρουν στους πελάτες τους κάποια μοντέλα που να παρουσιάζουν κάποιες ομοιότητες με τα χαρακτηριστικά του Ελεύθερου Λογισμικού. Το βασικό πρόβλημα που μπορεί να προκύψει με αυτό το μοντέλο, είναι ότι δημιουργεί σύγχυση στους πελάτες και στους developers, που θα πρέπει να διαβάσουν με λεπτομέρεια τα “μικρά γράμματα”, για να συνειδητοποιήσουν ότι αυτό που τους παρουσιάζεται δεν έχει όλα τα πλεονεκτήματα που το Ελεύθερο Λογισμικό θα μπορούσε να τους δώσει. Το πιο γνωστό τέτοιο μοντέλο είναι το πρόγραμμα “Shared Source” (Κοινού Κώδικα), της Microsoft.
- Έλλειψη γνώσης. Σε πολλές περιπτώσεις, οι χρήστες στρέφονται στο Ελεύθερο Λογισμικό απλώς επειδή νομίζουν πως είναι δωρεάν. Είτε, επειδή θεωρούν πως είναι “της μόδας”. Αν δεν το εξετάσουν περισσότερο σε βάθος, μελετώντας κάποιες σημαντικές λεπτομερειακές πληροφορίες για τα πλεονεκτήματα που το μοντέλο του Ελεύθερου Λογισμικού προσφέρει, τότε διατρέχουν τον κίνδυνο να μη μπορέσουν να επωφεληθούν πλήρως από τα πλεονεκτήματα αυτά. Σε πολλές περιπτώσεις, οι αρχικές παραδοχές, στον κόσμο του Ελεύθερου Λογισμικού, είναι τόσο διαφορετικές από τις παραδοσιακές παραδοχές που ισχύουν στον κόσμο του ιδιοταγούς λογισμικού, ώστε να απαιτείται μία ελάχιστη ανάλυση για να μπορούμε να κατανοήσουμε ότι εκείνο που μπορεί να είναι συχνό, στη μία περίπτωση, μπορεί, αντιθέτως, να είναι αδύνατο στην άλλη πλευρά, και αντίστροφα. Συνεπώς, η άγνοια μπορεί μόνο να οδηγήσει σε έλλειψη ικανοποίησης και σε απώλεια ευκαιριών για οποιονδήποτε, άτομο ή εταιρεία, που προσεγγίζει το Ελεύθερο Λογισμικό.
- Νομικά εμπόδια. Αυτό είναι, βεβαίως, το κύριο πρόβλημα που το Ελεύθερο Λογισμικό θα πρέπει να αντιμετωπίσει στα επόμενα χρόνια. Παρότι το νομικό περιβάλλον μέσα στο οποίο αναπτύχθηκε το Ελεύθερο Λογισμικό, στη δεκαετία του 80 και το πρώτο ήμισυ της δεκαετίας του 90, δεν ήταν ιδανικό, εν τούτοις του άφησε, τουλάχιστον, αρκετό χώρο για να μπορέσει να αναπτυχθεί ελεύθερα. Έκτοτε, η επέκταση της έννοιας των Διπλωμάτων Ευρεσιτεχνίας (*patenting*) στο Λογισμικό (που συνέβη σε πολλές ανεπτυγμένες χώρες), καθώς και η νέα νομοθεσία περί πνευματικών δικαιωμάτων (*copyright*), που περιορίζει την ελευθερία του developer Λογισμικού να δημιουργεί, διαμορφώνουν ολοένα και αυξανόμενα τείχη, που εμποδίζουν την διείσδυση του Ελεύθερου Λογισμικού σε σημαντικούς τομείς εφαρμογών.

2.6. Περίληψη

Αυτό το κεφάλαιο παρουσιάζει την ιστορία του Ελεύθερου Λογισμικού. Η δεκαετία του 1960 ήταν μία περίοδος όπου κυριαρχούσαν οι τεράστιοι υπολογιστές και η IBM, και κατά την οποία το Λογισμικό διενέμετο μαζί με το υλικό/ hardware, και, συνήθως, μαζί και με τον Πηγαίο Κώδικα. Μέσα στη δεκαετία του 1970, το Λογισμικό άρχισε να πωλείται χωριστά, και σύντομα εμφανίστηκαν ιδιοταγείς κλειστές διανομές, που έγιναν, σχεδόν, ο κανόνας και που δεν περιελάμβαναν τον Πηγαίο Κώδικα, ούτε παραχωρούσαν την άδεια τροποποίησης ή αναδιανομής. Κατά την δεκαετία του 1970, άρχισε να γίνεται δουλειά, με σκοπό την ανάπτυξη του Λειτουργικού Συστήματος Unix στα εργαστήρια της εταιρείας AT&T's Bell, το οποίο και, αργότερα, οδήγησε στην ανάπτυξη του Unix BSD. Η εξέλιξή του, παράλληλα με την γέννηση του Διαδικτύου, χρησίμευσε σαν ένα πεδίο δοκιμής για νέους τρόπους συνεργατικής ανάπτυξης, που στη συνέχεια, έγιναν πολύ κοινοί στον κόσμο του Ελεύθερου Λογισμικού.

Το 1984, ο Richard Stallman άρχισε να εργάζεται στο GNU project, ιδρύοντας το Free Software Foundation (FSF), γράφοντας την Άδεια GPL και, γενικότερα, καταξιώνοντας τα Ιδρύματα Ελεύθερου Λογισμικού με τον τρόπο που τα γνωρίζουμε σήμερα.

Μέσα στη δεκαετία του 90, το Διαδίκτυο ωρίμασε, προσφέροντας στις Κοινότητες Ελεύθερου Λογισμικού νέους διαύλους για την επικοινωνία και τη διανομή. Το 1991, ο Linus Torvalds άρχισε να αναπτύσσει ένα ελεύθερο πυρήνα (το Linux), ο οποίος και συνέδραμε στην ολοκλήρωση του συστήματος GNU, που είχε, ήδη, σχεδόν όλα τα επί μέρους τμήματα για να καταστεί ένα πλήρες λειτουργικό σύστημα, παρόμοιο με το Unix: τον μεταγλωττιστή C (GCC), τον editor (Emacs), το σύστημα διαχείρισης παραθύρων (X Window), κλπ.

Έτσι γεννήθηκαν τα λειτουργικά συστήματα GNU/Linux, διακλαδιζόμενα σε πολλές διαφορετικές διανομές, σαν την Red Hat Linux και την Debian GNU/Linux. Προς το τέλος της δεκαετίας του 90, αυτά τα συστήματα ολοκληρώθηκαν με δύο περιβάλλοντα για την επιφάνεια εργασίας: το KDE και το GNOME.

Κατά την δεκαετία του 2000, το Ελεύθερο Λογισμικό κατάφερε να κατακτήσει ηγετική θέση σε μερικούς τομείς (όπως ο τομέας των Διαδικτυακών διακομιστών, που εκυριαρχείτο από τον Apache) και εμφανιζόταν νέα εργαλεία που κάλυπταν ένα μεγάλο αριθμό απαιτήσεων Πληροφορικής (IT).

Δείτε επίσης:

Όσοι αναγνώστες ενδιαφέρονται, μπορούν να βρουν στο Παράρτημα Β, έναν κατάλογο με μερικές από τις σημαντικότερες ημερομηνίες στην ιστορία του Ελεύθερου Λογισμικού.

3. Νομικές πλευρές

"Οι άδειες για τα περισσότερα λογισμικά είναι σχεδιασμένες με τέτοιο τρόπο ώστε να σας στερούν την ελευθερία να μπορείτε να τα μοιράζεστε με άλλους και να τα τροποποιείτε".

GNU General Public Licence, version 2

Αυτό το κεφάλαιο εξετάζει τις κυριότερες νομικές πλευρές που σχετίζονται με το Ελεύθερο Λογισμικό. Για να τις εντάξουμε σε ένα πλαίσιο, αρχίζουμε με μία μικρή εισαγωγή στις πιο βασικές έννοιες περί δικαιωμάτων πνευματικής και βιομηχανικής ιδιοκτησίας (*intellectual and industrial*

property), πριν παρουσιάσουμε τον αναλυτικό ορισμό του *Ελεύθερου Λογισμικού*, του *Λογισμικού Ανοικτού Κώδικα* (*open source*) και άλλων παρεμφερών εννοιών. Θα εξετάσουμε, επίσης, αρκετά σε βάθος, τις πιο συνηθισμένες Άδειες λογισμικού, καθώς και την επιρροή τους στα επιχειρηματικά μοντέλα (αντικείμενο που καλύπτεται πιο αναλυτικά στο κεφάλαιο 5 και στα μοντέλα ανάπτυξης).

3.1. Σύντομη εισαγωγή στην πνευματική ιδιοκτησία

Ο όρος *πνευματική ιδιοκτησία* προσλαμβάνει διαφορετικές σημασίες, ανάλογα με το πλαίσιο και το ποιος τον χρησιμοποιεί. Στις μέρες μας, χρησιμοποιείται συχνά σε πολλούς τομείς, για να αναφερθούμε στα διάφορα προνόμια που παραχωρούνται σε άυλα αγαθά, που έχουν οικονομική αξία. Περιλαμβάνει έννοιες όπως το *copyright* και άλλες παρόμοιες, που παρέχουν προστασία εναντίον των *μη-εγκριμμένων αντιγράφων*, στην περίπτωση των λογοτεχνικών έργων, των καλλιτεχνικών έργων, των προγραμμάτων για υπολογιστή, των συλλογών δεδομένων, των βιομηχανικών σχεδίων, των κατατεθειμένων εμπορικών σημάτων (*trademarks*), που προστατεύουν τα σύμβολα, των γεωγραφικών προσδιορισμών, που προστατεύουν τις *Ονομασίες Ελεγχμένης Προέλευσης*, των εμπορικών μυστικών, που προστατεύουν την απόκρυψη πληροφοριών, και των διπλωμάτων ευρεσιτεχνίας (*patents*), που παρέχουν δικαιώματα προσωρινού μονοπωλίου σε διάφορες εφευρέσεις, σε αντάλλαγμα για την αποκάλυψή τους. Εν πάσει περιπτώσει, στην παράδοση πολλών νομικών συστημάτων, περιλαμβανομένης της Ισπανικής παράδοσης, υιοθετείται μία διάκριση ανάμεσα στην *πνευματική ιδιοκτησία*, που αναφέρεται αποκλειστικά στην *προστασία πρωτότυπων δημιουργημάτων λόγου, τέχνης ή επιστήμης έργων* (άρθρο 2 παρ. 1 Ν.2121/1993), και την λεγόμενη *βιομηχανική ιδιοκτησία* (*industrial property*), που καλύπτει όλες τις άλλες έννοιες.

Σε κάθε περίπτωση, η νομοθεσία που εφαρμόζεται σε όλες αυτές τις περιπτώσεις, είναι μία από τις καλύτερα συντονισμένες, σε πρακτικό επίπεδο, διεθνώς. Από τη μία πλευρά, η WIPO (Διεθνής Παγκόσμια Οργάνωση Ιδιοκτησίας) καλύπτει και τις δύο αυτές μορφές ιδιοκτησίας, σε όλες της τις εκφάνσεις. Από την άλλη πλευρά, η συμφωνία TRIPS (Trade-Related aspects of Intellectual Property rights) καθορίζει ορισμένα ελάχιστα επίπεδα προστασίας και υποχρεώνει όλες τις χώρες-μέλη του Π.Ο.Ε. (Παγκόσμιου Οργανισμού Εμπορίου/ WTO, World Trade Organisation) να τα ενσωματώσουν, εντός κάποιων συγκεκριμένων χρονοδιαγραμμάτων, αναλόγως του βαθμού ανάπτυξης της κάθε χώρας.³

(³) Το σύμφωνο TRIPS υπεγράφη υπό καθεστώς πίεσης, εκ μέρους των βιομηχανοποιημένων χωρών (ειδικά των ΗΠΑ και της Ιαπωνίας).

Το Άρθρο 27 της Δήλωσης των Ανθρωπίνων Δικαιωμάτων, αναγνωρίζει ότι ο καθένας έχει το δικαίωμα στην προστασία των ηθικών και υλικών συμφερόντων που απορρέουν από οποιοδήποτε επιστημονικό, λογοτεχνικό, ή καλλιτεχνικό έργο, του οποίου είναι ο δημιουργός/ συγγραφέας.

Πάντως, σε πολλές περιπτώσεις (και συχνά στην περίπτωση του Λογισμικού), αυτό το δικαίωμα μεταφέρεται, πρακτικά, στις εταιρείες που απασχολούν τους δημιουργούς, ή που διανέμουν, ή που πωλούν τα έργα τους. Παρ' όλα ταύτα, η πνευματική ιδιοκτησία δικαιολογείται όχι μόνο σε ηθικό επίπεδο, αλλά και για πρακτικούς λόγους, ούτως ώστε να υπάρχει συμμόρφωση και με ένα άλλο δικαίωμα: το δικαίωμα του ευρέος κοινού να μπορεί να αντλεί όφελος από τη δημιουργία, προωθώντας την μέσω παροχής κινήτρων, και προστατεύοντας τις επενδύσεις στον τομέα της δημιουργίας, της έρευνας και της ανάπτυξης. Για να μπορέσουν να εναρμονισθούν αυτά τα δύο δικαιώματα, η πνευματική ιδιοκτησία είναι παροδικής φύσεως και λήγει άπαξ και εκπληρώσει το ρόλο της στο θέμα της προώθησης.

Αλλά η έννοια της λήξης, δεν αποτελεί το μοναδικό διακριτικό γνώρισμα ανάμεσα στην πνευματική ιδιοκτησία και την κανονική ιδιοκτησία. Στην εποχή μας, αυτά τα προϊόντα μπορούν να αντιγραφούν εύκολα και φθηνά, χωρίς καμία απώλεια ποιότητας. Η αντιγραφή δεν βλάπτει (*prejudice*) εκείνον που ήδη αντλεί οφέλη από το αντιγραφόμενο έργο, σε αντίθεση με την κλοπή, η οποία το στερεί από τον αρχικό ιδιοκτήτη. Η αντιγραφή μπορεί να βλάψει τον ιδιοκτήτη, στερώντας του την άντληση, εν δυνάμει, εσόδων που θα προερχόταν από την πώληση. Ο έλεγχος των άυλων έργων είναι πολύ πιο περίπλοκος από τον έλεγχο της κλοπής υλικής ιδιοκτησίας και μπορεί να μας οδηγήσει σε ένα αστυνομοκρατούμενο κράτος, με την υποχρέωση ελέγχου όλων των αντιγράφων κάθε πληροφορίας, αλλά και σε καθεστώς νομικής ανασφάλειας, από την στιγμή που η πιθανότητα για τυχαία παραβίαση αυτών των δικαιωμάτων, αυξάνει. Πέραν αυτού, η δημιουργικότητα είναι προσ αυξητική (*incremental*): η πράξη της δημιουργίας, πάντοτε αντιγράφει κάτι, και η διαχωριστική γραμμή ανάμεσα σε μία φτωχή απομίμηση και την έμπνευση, είναι πολύ λεπτή.

Για να μπορέσουμε να το μελετήσουμε αυτό, σε μεγαλύτερο βάθος, στα επόμενα κεφάλαια θα ανασκοπήσουμε μερικές κατηγορίες πνευματικής ιδιοκτησίας. Σε κάθε περίπτωση, μπορούμε ήδη να προκαταβάλλουμε ότι το Ελεύθερο Λογισμικό προτείνει ένα σημείο ισορροπίας σε αυτό τον τομέα, υπερασπιζόμενο τα πλεονεκτήματα της αντιγραφής και της προσ αυξητικής καινοτομίας (*incremental innovation*), σε αντιπαράθεση με τον αποκλειστικό έλεγχο ενός έργου εκ μέρους του συγγραφέως.

3.1.1. To Copyright

Το Copyright προστατεύει την έκφραση ενός περιεχομένου, αλλά όχι το ίδιο το περιεχόμενο. Το Copyright αναπτύχθηκε για να ανταμείψει τους συγγραφείς βιβλίων ή έργων τέχνης. Τα προστατευμένα έργα μπορεί να εκφράζουν ιδέες, γνώση ή μεθόδους, που είναι ήδη σε ελεύθερη χρήση, αλλά απαγορεύεται η αναπαραγωγή τους χωρίς πλήρη ή μερική άδεια, με ή χωρίς αλλαγές. Αυτή η προστασία είναι πολύ απλή, αφού εφαρμόζεται αυτομάτως, σχεδόν σε διεθνή συνολική κλίμακα, αμέσως μόλις το έργο δημοσιευθεί/ εκδοθεί. Έχει επεκταθεί τρεχόντως και σε προγράμματα υπολογιστών, αλλά και (σε μερικές γεωγραφικές περιοχές) σε συλλογές δεδομένων.

Ο Νόμος περί Πνευματικής Ιδιοκτησίας στην Ισπανία, καθώς και οι παρόμοιοι Νόμοι σε άλλα κράτη, αναπτύχθηκαν βασιζόμενοι στη Συνθήκη της Βέρνης του 1886, που προοριζόταν για την προστασία των λογοτεχνικών έργων και των έργων τέχνης, και ρυθμίζει το copyright. Αυτά τα δικαιώματα διακρίνονται στο Ηθικό και στο Περιουσιακό δικαίωμα. Το πρώτο εγγυάται τον έλεγχο του συγγραφέως επί της διανομής του έργου του, με το όνομά του ή με ένα ψευδώνυμο, την αναγνώριση της προέλευσης της συγγραφής (*authorship*), το σεβασμό για την ακεραιότητα του έργου και το δικαίωμα μετατροπής του ή και της απόσυρσής του. Το δεύτερο, δίνει στον συγγραφέα το δικαίωμα να εκμεταλλευθεί οικονομικά το έργο και να μπορεί να το παραχωρήσει συνολικά, ή εν μέρει, κατ' αποκλειστικότητα ή όχι, σε έναν τρίτο ενδιαφερόμενο.

Τα πνευματικά δικαιώματα (ηθικό και περιουσιακό) ισχύουν μέχρι και εβδομήντα (70) έτη μετά την αποβίωση του δημιουργού στις περισσότερες χώρες, όπως και στην Ελλάδα

Η παραχώρηση αυτών των δικαιωμάτων σε προγράμματα λογισμικού καθορίζεται μέσω ενός συμβολαίου γνωστού ως “*Άδεια*”. Στην περίπτωση ιδιοταγών, κλειστών λογισμικών, αυτά διανέμονται προς χρήση με “μη αποκλειστικές” άδειες, με την αντίληψη ότι γίνονται αποδεκτές αυτομάτως, με το που ο χρήστης ανοίγει ή εγκαθιστά το προϊόν. Για να ισχύσει η προστασία που απορρέει από την πνευματική ιδιοκτησία, δεν είναι, άρα, απαραίτητο να υπογραφεί το συμβόλαιο, αφού στην περίπτωση ενός αποδέκτη που δεν το αποδέχεται, τότε και πάλι θα ισχύει αυτομάτως η

προστασία των δικαιωμάτων πνευματικής ιδιοκτησίας που προβλέπονται από τον Νόμο, πράγμα που σημαίνει κανένα δικαίωμα. Οι Άδειες δεν μπορούν να περιορίσουν μερικά από τα δικαιώματα που παραχωρεί η τρέχουσα νομοθεσία, όπως το δικαίωμα αναπαραγωγής ιδιωτικών αντιτύπων έργων τέχνης ή μουσικής, πράγμα που επιτρέπει να φτιαχτεί ένα αντίγραφο ενός δίσκου μουσικής και να δωθεί ως δώρο σε ένα φίλο, αλλά αυτό το δικαίωμα δεν βρίσκει εφαρμογή στα προγράμματα υπολογιστών. Σύμφωνα με το LPI του 1996 (Ισπανικός Νόμος περί Πνευματικής Ιδιοκτησίας. Βασιλικό Νομοθετικό Διάταγμα 1/1996, της 12ης Απριλίου) [77], όπως τροποποιήθηκε το 2006 (Νόμος περί Πνευματικής Ιδιοκτησίας. Νόμος 23/2006, της 7ης Ιουλίου) [79], σχετικά με τα προγράμματα, παραμένει πάντα η δυνατότητα να παίρνουμε ένα αντίγραφο ασφαλείας, να μπορούν να μελετηθούν για να καταστούν διαλειτουργικά και να μπορούν να διορθωθούν και να τροποποιηθούν σύμφωνα με τις ανάγκες μας (πράγμα που είναι πολύ δύσκολο, διότι, συνήθως, δεν έχουμε τρόπο πρόσβασης στον Πηγαίο Κώδικα). Αυτά τα δικαιώματα δεν μπορούν να περιοριστούν αρκετά μέσω των αδειών, αν και οι Νόμοι βρίσκονται υπό αναθεώρηση, σε μία προφανώς ασταμάτητη τάση περιορισμού των δικαιωμάτων των χρηστών. Οι οργανωμένες συλλογές έργων ή δεδομένων τρίτων, υπόκεινται επίσης σε *προστασία δικαιωμάτων πνευματικής ιδιοκτησίας (copyright)*, αν και υπό διαφορετικούς όρους, με στενότερα χρονικά πλαίσια.

Οι νέες τεχνολογίες της πληροφορικής, και ειδικά το Διαδίκτυο, μεταμόρφωσαν βαθιά την προστασία της πνευματικής ιδιοκτησίας, από τη στιγμή που είναι πολύ ευκολότερο να αντιγράψει κανείς τους τρόπους έκφρασης περιεχομένου, παρά το περιεχόμενο αυτό καθ' εαυτό. Και στην περίπτωση των προγραμμάτων και ορισμένων έργων τέχνης (μουσική, εικόνες, κινηματογραφικές ταινίες, ακόμη και λογοτεχνικά έργα) μπορούν να “λειτουργήσουν” αυτόματα στον υπολογιστή χωρίς την ανάγκη καμίας ιδιαίτερης ανθρώπινης προσπάθειας. Παρ' όλα ταύτα, διάφορα σχέδια ή εφευρέσεις πρέπει να φτιαχθούν και, ενδεχομένως, να μουν και στην παραγωγή. Αυτή η δυνατότητα δημιουργίας πλούτου χωρίς κανένα κόστος, οδήγησε ένα μεγάλο μέρος του κοινού, και ειδικότερα στις φτωχές χώρες, να αντιγράφουν προγράμματα χωρίς να πληρώνουν για την άδεια και χωρίς να υπάρχει δημόσια συνειδητοποίηση (*public awareness*) για το ότι πρόκειται περί “κακόβουλης πράξης” (αντίθετα με την περίπτωση κλοπής μίας φυσικής ιδιοκτησίας, για παράδειγμα). Στο εν τω μεταξύ, οι δημιουργοί προγραμμάτων, είτε μόνοι τους, είτε σε συνασπισμό (για παράδειγμα, μέσω της BSA, Business Software Alliance), ασκούν τεράστια πίεση για να πληρώνονται οι άδειες και για να κυνηγούν οι κυβερνήσεις αυτό που έγινε γνωστό ως πειρατεία.

Σημείωση

Η λέξη πειρατεία έγινε γενικώς αποδεκτή ως συνώνυμο του όρου “παραβίαση οποιασδήποτε μορφής πνευματικής ιδιοκτησίας, ειδικά στην περίπτωση παράνομης αντιγραφής προγραμμάτων, μουσικής και ταινιών”. Ο όρος μοιάζει υπερβολικός και, στο Λεξικό της Βασιλικής Ισπανικής Ακαδημίας της Γλώσσας, εμφανίζεται με την συγκεκριμένη ερμηνεία κατά την μεταφορική έννοια, αφού η αρχική λέξη αναφέρεται σε “ληστεία μετά βίας, που λαμβάνει χώρα στη θάλασσα”. Αυτός είναι και ο λόγος για τον οποίο ο Richard Stallman συστήνει να αποφεύγεται (“Some confusing or loaded words and phrases that are worth avoiding”, 2003) [212].

Ακριβώς για να προστατευθεί το copyright των περιεχομένων με ιδιοταγείς άδειες, προέκυψαν τα λεγόμενα συστήματα DRM (“*Digital Rights Management*”/ “*Διαχείριση Ψηφιακών Δικαιωμάτων*”), που σχεδιάστηκαν για να ελέγχουν την πρόσβαση και τη χρήση των δεδομένων σε ψηφιακή μορφή, ή για να περιορίσουν τη χρήση τους σε ορισμένες συσκευές. Η χρήση των συστημάτων DRM δέχθηκε βαριά κριτική σε πολλά επίπεδα, διότι προστατεύουν το copyright, επιβάλλοντας περιορισμούς πέραν του επαρκούς, πράγμα που εξηγεί γιατί μερικοί, σαν το Free Software Foundation, συστήνουν την ερμηνεία του ακρώνυμου ως “*Digital Restrictions Management*”/ “*Διαχείριση Ψηφιακών Περιορισμών*”, σε μία προσπάθεια να αποφευχθεί η χρήση της λέξης “*δικαιώματα*”, διότι θεωρεί ότι υπάρχει μία υπερβολική αποστέρηση των δικαιωμάτων των χρηστών, προς χάριν της ικανοποίησης των απαιτήσεων του copyright.

3.1.2. Εμπορικό Μυστικό

Μία άλλη πηγή που οι εταιρείες εκμεταλλεύονται για να πραγματοποιήσουν κέρδη από τις επενδύσεις τους, είναι το εμπορικό μυστικό, που προστατεύεται από τους νόμους περί βιομηχανικής ιδιοκτησίας, υπό την προϋπόθεση ότι οι εταιρείες θα παίρνουν επαρκή μέτρα για την απόκρυψη των πληροφοριών που δεν θέλουν να αποκαλύψουν. Στην περίπτωση χημικών ή φαρμακευτικών προϊόντων, για τα οποία απαιτείται κυβερνητική έγκριση, η κυβέρνηση αναλαμβάνει να μην αποκαλύπτει τα υποβαλλόμενα δεδομένα, τα οποία δεν είναι απαραίτητο να δημοσιεύονται.

Μία από τις καλύτερα γνωστές εφαρμογές του εμπορικού μυστικού είναι η βιομηχανία του ιδιοταγούς, κλειστού λογισμικού, η οποία, σε γενικές γραμμές, πωλεί μεταγλωττισμένα (*compiled*) προγράμματα, χωρίς πρόσβαση στον πηγαίο κώδικα, για να αποφεύγεται η ανάπτυξη παραγώγων προγραμμάτων.

Σε πρώτη ματιά, η προστασία του εμπορικού μυστικού μοιάζει σαν κάτι το διεστραμμένο, αφού μπορεί να αποστερεί στο διηνεκές τη διάδοση χρήσιμης γνώσης στην κοινωνία. Σε κάποιο βαθμό, ακόμη και μερικές νομοθεσίες το ερμηνεύουν κατ' αυτό τον τρόπο, επιτρέποντας τις τεχνικές ανάστροφης μηχανικής (*reverse engineering*), για να μπορούν να αναπτύσσονται προϊόντα αντικατάστασης, αν και με την πίεση που ασκεί η βιομηχανία κατάφερε να πετύχει την απαγόρευση αυτής της δραστηριότητας σε πολλές χώρες, ενώ σε άλλες χώρες επιτράπη μόνον με την λογική της συμβατότητας.

Ανεξαρτήτως του κατά πόσο το εμπορικό μυστικό είναι ή όχι διαστροφή, σε πολλές περιπτώσεις είναι καλύτερο από ένα δίπλωμα ευρεσιτεχνίας (*patent*), καθότι προσφέρει ένα ανταγωνιστικό πλεονέκτημα σε εκείνον που προωθεί ένα προϊόν στην αγορά, ενώ οι ανταγωνιστές προσπαθούν να το μιμηθούν μέσω της ανάστροφης μηχανικής. Όσο πιο εξεζητημένο είναι ένα προϊόν, τόσο περισσότερο θα κοστίζει στους ανταγωνιστές να το αναπαραγάγουν, ενώ εάν είναι ασήμαντο (*trivial*), θα αντιγραφεί τάχιστα. Η απομίμηση, μαζί με τις βελτιώσεις, υπήρξαν βασικής σημασίας στην ανάπτυξη των σημερινών υπερδυνάμεων (των ΗΠΑ και της Ιαπωνίας) και είναι πολύ σημαντικές για την οικονομική ανεξαρτησία των αναπτυσσόμενων χωρών.

3.1.3. Διπλώματα Ευρεσιτεχνίας (*Patents*) και μοντέλα *utility*

Η εναλλακτική στην έννοια του εμπορικού μυστικού είναι το δίπλωμα ευρεσιτεχνίας (*patent*). Σε αντάλλαγμα για ένα μονοπώλιο 17-25 ετών και ένα συγκεκριμένο χρηματικό κόστος, μία εφεύρεση αποκαλύπτεται δημοσίως, για να μπορεί εύκολα να αναπαραχθεί. Στοχεύει στην προώθηση της ιδιωτικής έρευνας, χωρίς κανένα κόστος για τον φορολογούμενο και χωρίς να χαθεί το αποτέλεσμα. Ο κάτοχος του Διπλώματος Ευρεσιτεχνίας μπορεί να αποφασίσει αν θα επιτρέψει σε άλλους να το χρησιμοποιήσουν καθώς και να ορίσει το τίμημα που θα πρέπει να καταβληθεί για την Άδεια.

Το επίσημο δόγμα είναι ότι το σύστημα διπλωμάτων ευρεσιτεχνίας προωθεί την καινοτομία, αλλά όλο και περισσότερες φωνές υψώνονται υπέρ της άποψης ότι την εμποδίζει, είτε διότι πρόκειται για ένα κακοεφαρμοσμένο σύστημα, είτε διότι θεωρούν το σύστημα αυτό, καθ' εαυτό, ως διεστραμμένο (*François-René Rideau, "Patents are an economic absurdity", 2000*) [194].

Το τι θεωρείται ως εφεύρεση (*invention*) αλλάζει καθώς περνά ο καιρός, και ασκείται τεράστια πίεση για να επεκταθεί περαιτέρω η ισχύς του συστήματος, περιλαμβάνοντας αλγόριθμους, προγράμματα, επιχειρηματικά μοντέλα, φυσικές ουσίες, γονίδια και μορφές ζωής,

περιλαμβανομένων φυτών και ζώων. Το σύμφωνο TRIPS απαιτεί να μην γίνεται διάκριση εναντίον οποιουδήποτε πεδίου γνώσης, στο πλαίσιο του συστήματος των διπλωμάτων ευρεσιτεχνίας. Οι πιέσεις του Παγκόσμιου Οργανισμού Πνευματικής Ιδιοκτησίας (*World Intellectual Property Organisation, WIPO*) αποσκοπούν να εξαφανίσουν την ανάγκη για μία εφεύρεση να βρίσκει και μία βιομηχανική εφαρμογή, αλλά και στο να μειώσουν τις προδιαγραφές που απαιτούνται για να πάρουν οι εφευρέσεις ένα δίπλωμα ευρεσιτεχνίας. Οι ΗΠΑ βρίσκονται στην πρωτοπορία των χωρών με ελάχιστες απαιτήσεις για την απόκτηση διπλώματος ευρεσιτεχνίας (*patentability*), και είναι και οι πιο επιθετικές στο να επιβάλλουν στις άλλες χώρες τις δικές τους προδιαγραφές, ξεχνώντας ότι οι ΗΠΑ ήταν εκείνες που αρνούνταν να αποδεχθούν τα ξένα Διπλώματα Ευρεσιτεχνίας, όταν εκείνες ήταν ακόμη μία αναπτυσσόμενη χώρα.

Μετά την απόκτηση ενός Διπλώματος Ευρεσιτεχνίας (*patent*), τα δικαιώματα του κατόχου είναι ανεξάρτητα από την ποιότητα της εφεύρεσης και από την προσπάθεια που επενδύθηκε για την απόκτησή του. Με δεδομένο το κόστος της διατήρησης ενός Διπλώματος και του κόστους των μηνύσεων, μόνον οι μεγάλες εταιρείες καταφέρνουν να τα διατηρούν και, πράγματι, διατηρούν ένα τεράστιο portfolio Διπλωμάτων ευρεσιτεχνίας, που τις φέρνει σε θέση να μπορούν να στραγγαλίζουν οιονδήποτε ανταγωνισμό. Με δεδομένη την ευκολία με την οποία αποκτά κανείς ένα Δίπλωμα ακόμη και για ταπεινές ή ευρέως εφηρμοσμένες λύσεις, μπορεί να μονοπωλεί ολόκληρους τομείς της οικονομικής δραστηριότητας.

Με τα Διπλώματα ευρεσιτεχνίας (*patents*), πολλές δραστηριότητες, και ειδικά ο προγραμματισμός, καθίστανται εξαιρετικά επικίνδυνες, καθώς είναι πανεύκολο να υπάρξει μία τυχαία παραβίαση κάποιου διπλώματος ευρεσιτεχνίας (*patent*), κατά την διαδικασία ανάπτυξης ενός σύνθετου προγράμματος. Όταν δύο ή και περισσότερες εταιρείες διεξάγουν μία έρευνα για την επίλυση ενός προβλήματος, είναι πολύ πιθανό πως θα καταλήξουν, σχεδόν συγχρόνως, σε μία παρόμοια λύση, αλλά μόνο μία από αυτές (συνήθως εκείνη με τους μεγαλύτερους χρηματικούς πόρους) θα καταφέρει να πατεντάρει την εφεύρεσή της, αποκλείοντας κάθε πιθανότητα να μπορέσουν ποτέ οι άλλες εταιρείες να ανακτήσουν την επένδυσή τους. Κάθε σύνθετη τεχνολογική πρόοδος μετατρέπεται σε εφιάλη, αν για κάθε προσπάθεια επίλυσης ενός προβλήματος, πρέπει πρώτα να ανακαλύψει κανείς κατά πόσον η ανακαλυφθείσα λύση είναι ήδη πατενταρισμένη (ή αν τελεί εν αναμονή έκδοσης Διπλώματος ευρεσιτεχνίας), για να μπορέσει να αποκτήσει την άδεια ή να για να βρει μία εναλλακτική λύση. Αυτό το πρόβλημα είναι ιδιαίτερα σοβαρό με το Ελεύθερο Λογισμικό, όπου η παραβίαση των πατεντών αλγορίθμων είναι προφανής με μία απλή επισκόπηση του Κώδικα. Παρ' ότι στην Ευρώπη παραμένει, προς το παρόν, παράνομη η έκδοση Διπλώματος Ευρεσιτεχνίας για έναν αλγόριθμο, αυτό ίσως καταστεί εφικτό στο εγγύς μέλλον, ακόμη και την στιγμή που ο αναγνώστης διαβάζει αυτό το βιβλίο.

3.1.4. Εμπορικά Σήματα Κατατεθέντα (*Registered trademarks*) και Λογότυπα (*logos*)

Τα Εμπορικά Σήματα Κατατεθέντα και τα Λογότυπα αποτελούν ονόματα και σύμβολα που αντιπροσωπεύουν μία καθιερωμένη ποιότητα (ή μία μαζική επένδυση σε διαφήμιση). Δεν είναι κάτι το ιδιαίτερα σημαντικό στον κόσμο του Ελεύθερου Λογισμικού, ίσως επειδή η καταχώρησή τους συνεπάγεται και ένα κόστος.

Συνεπώς, μόλις λίγα σημαντικά ονόματα, όπως “Ανοικτός Κώδικας” / “Open Source” (από το Ίδρυμα Open Source Foundation), το Debian (από το Software in the Public Interest), το GNOME (από το Ίδρυμα GNOME Foundation), το GNU (από το Ίδρυμα Free Software Foundation) ή το OpenOffice.org (από την SUN Microsystems) είναι κατατεθειμένα, και μόνο σε λίγες χώρες. Πάντως, η μη καταχώρηση των ονομάτων δημιούργησε προβλήματα. Για παράδειγμα, στις ΗΠΑ

(το 1996) και στην Κορέα (το 1997), κάποιοι άνθρωποι καταχώρησαν το όνομα “Linux” και απαίτησαν πληρωμή για την χρήση του. Η επίλυση αυτών των προστριβών επιφέρει νομικά κόστη και την ανάγκη να αποδειχθεί η χρήση του ονόματος πριν από την ημερομηνία της καταχώρησης.

3.2. Άδειες Ελεύθερου Λογισμικού

Νομικά μιλώντας, η κατάσταση των ελεύθερων προγραμμάτων σε σχέση με τα ιδιοταγή κλειστά λογισμικά, δεν είναι πολύ διαφορετική: αμφότερα διανέμονται με μία Άδεια. Η διαφορά έγκειται στο τι ακριβώς επιτρέπει αυτή η άδεια. Στην περίπτωση των αδειών των ελεύθερων προγραμμάτων, οι οποίες δεν περιορίζουν ιδιαίτερα την χρήση τους, την αναδιανομή και την τροποποίησή τους, αυτό που μπορεί να επιβληθεί είναι ποιες συνθήκες πρέπει να πληρούνται με ακρίβεια, στην περίπτωση που θέλουμε να αναδιανείμουμε το πρόγραμμα. Για παράδειγμα, είναι δυνατόν να απαιτηθούν ενδείξεις περί των δημιουργών (*authorship indications*) ή να συμπεριλαμβάνει ο πηγαίος κώδικας, εάν επιθυμούμε, την αναδιανομή ενός προγράμματος που είναι έτοιμο να τρέξει. Αν και, ουσιαστικά, το *Ελεύθερο Λογισμικό* και το *ιδιοταγές κλειστό λογισμικό*, διαφέρουν στους όρους της Αδείας υπό την οποία οι δημιουργοί κυκλοφορούν τα προγράμματά τους, είναι σημαντικό να τονίσουμε ότι αυτή η διάκριση αντανακλάται και σε πλήρως διαφορετικούς όρους χρήσης και αναδιανομής. Όπως διαπιστώσαμε κατά τα τελευταία χρόνια, αυτό δεν οδήγησε μόνον σε εντελώς διαφορετικές μεθόδους ανάπτυξης, αλλά και σε τρόπους πρακτικώς αντίθετους (από πολλές απόψεις) σε σχέση με την κρατούσα αντίληψη για την Τεχνολογία της Πληροφορίας (IT).

Οι νόμοι περί Πνευματικής Ιδιοκτησίας εξασφαλίζουν ότι όταν δεν υπάρχει μία ξεκάθαρη άδεια, τότε δεν μπορούμε να κάνουμε απολύτως τίποτε με ένα έργο (στην περίπτωσή μας, με ένα πρόγραμμα), που είτε μας δόθηκε, είτε το αγοράσαμε. Μόνον ο συγγραφέας (ή ο κάτοχος των δικαιωμάτων του έργου) μπορεί να μας παραχωρήσει αυτή την άδεια. Σε κάθε περίπτωση, η κατοχή ενός έργου δεν αλλάζει με την παραχώρηση μίας αδείας χρήσεως, αφού αυτή δεν επιφέρει μεταφορά της ιδιοκτησίας, παρά μόνον το δικαίωμα χρήσης, και σε κάποιες περιπτώσεις (υποχρεωτικά για το Ελεύθερο Λογισμικό) και το δικαίωμα διανομής και τροποποίησης. Οι Άδειες Ελεύθερου Λογισμικού διαφέρουν από εκείνες του ιδιοταγούς, κλειστού λογισμικού, ακριβώς κατά το ότι, αντί να περιορίζουν προσεκτικά αυτά που επιτρέπονται, προχωρούν στο να κάνουν κάποιες σαφείς παραχωρήσεις. Όταν οι άνθρωποι αποδέχονται ένα Ελεύθερο Πρόγραμμα μπορούν να αποφασίσουν να το αναδιανείμουν ή όχι, αλλά εάν το πράξουν, μπορούν να το κάνουν διότι η άδεια το επιτρέπει. Για να γίνει, όμως, αυτό, θα πρέπει η άδεια να γίνεται σεβαστή. Πράγματι, η άδεια περιέχει τους κανόνες χρήσης που θα πρέπει να σεβαστούν οι πάντες, οι χρήστες, οι αναδιανομείς, εκείνοι που ενοποιούν εφαρμογές (*integrators*) και όλοι οι άλλοι ενδιαφερόμενοι στον κόσμο IT.

Για να κατανοήσουμε όλες τις νομικές λεπτομέρειες αυτού του κεφαλαίου, “με το νι και με το σίγμα” και οι οποίες είναι, αναμφισβήτητα, πολύ σημαντικές για να καταλάβουμε την φύση του Ελεύθερου Λογισμικού, θα πρέπει επίσης να έχουμε υπ’ όψιν ότι κάθε νέα έκδοση του προγράμματος θεωρείται ως ένα νέο έργο. Ο δημιουργός του μπορεί, για μία ακόμη φορά, να κάνει ό,τι θέλει με αυτό, ακόμη και να το διανείμει υπό εντελώς διαφορετικούς όρους και προϋποθέσεις (με άλλα λόγια, με μία εντελώς διαφορετική άδεια, σε σχέση με την προηγούμενη). Κατ’ αυτόν τον τρόπο, εάν ο αναγνώστης είναι ο μόνος συγγραφέας ενός προγράμματος, μπορεί να δημοσιεύσει μία έκδοση υπό μία άδεια Ελεύθερου Λογισμικού και, αν επιλέξει διαφορετικά, να κυκλοφορήσει μία κατοπινή έκδοση υπό μία άδεια ιδιοταγούς λογισμικού. Στην περίπτωση ύπαρξης περισσότερων συγγραφέων και, εφ’ όσον η νέα έκδοση περιέχει κώδικα που αυτοί παρήγαγαν, εάν ο κώδικας πρόκειται να δημοσιευθεί υπό διαφορετικές συνθήκες, τότε όλοι οι συγγραφείς θα πρέπει να εγκρίνουν την αλλαγή Αδείας. Ένα θέμα που παραμένει ακόμη ανοιχτό, είναι ο τύπος της Αδείας που εφαρμόζεται για τις εξωτερικές συνεισφορές.

Σε γενικές γραμμές, θεωρείται ότι όταν κάποιος συνεισφέρει σε ένα project, τότε αποδέχεται de facto ότι η συνεισφορά του/της θα ενταχθεί στους όρους που προβλέπονται από την Άδεια, αν και το νομικό υπόβαθρο αυτής της παραδοχής παραμένει ισχύον. Η διαπίστωση της πρωτοβουλίας του Ιδρύματος για το Ελεύθερο Λογισμικό να ζητά, μέσω μίας γραπτής επιστολής, την παραχώρηση κάθε δικαιώματος copyright, από οποιονδήποτε συνεισφέρει άνω των 10 γραμμών κώδικα σε ένα GNU sub-project, συνιστά μία ξεκάθαρη ένδειξη ότι στον κόσμο του Ελεύθερου Λογισμικού ισχύουν αυστηρότερες τακτικές (*policies*) σε σχέση με αυτές τις συνεισφορές.

Με βάση τα παραπάνω, στο υπόλοιπο κομμάτι αυτού του κεφαλαίου θα εστιάσουμε στην ανάλυση διαφόρων αδειών. Για να εντάξουμε αυτή την ανάλυση σε ένα πλαίσιο, πρέπει να θυμόμαστε ότι από τώρα και στο εξής, όταν λέμε ότι μία Άδεια είναι Άδεια Ελεύθερου Λογισμικού, βασικά εννοούμε ότι οι πληροί τους ορισμούς για το Ελεύθερο Λογισμικό που παρουσιάστηκαν στο Κεφάλαιο 1.1.1.

3.2.1. Τύποι Αδειών

Υπάρχει μία τεράστια ποικιλία Ελεύθερων Αδειών, αν και για πρακτικούς λόγους τα περισσότερα projects χρησιμοποιούν ένα μικρό σύνολο τεσσάρων ή πέντε Αδειών. Από την άλλη μεριά, πολλά projects δεν επιθυμούν ή δεν έχουν την δυνατότητα να σχεδιάσουν τη δική τους άδεια. Από την άλλη πλευρά, οι περισσότεροι χρήστες προτιμούν να έχουν να κάνουν με μία πολύ γνωστή Άδεια, παρά να χρειάζεται να διαβάσουν και να αναλύσουν εξ' ολοκλήρου νέες άδειες.

Βιβλιογραφία

Στο άρθριο "Free licences" [121] του Ιδρύματος Ελεύθερου Λογισμικού, υπάρχει μία ανασκόπηση και η συζήτηση για τις άδειες που θεωρούνται Μη-Ελεύθερες ή Ελεύθερες, αλλά μη συμβατές με τη Γενική δημόσια Άδεια GPL, από την σκοπιά του FSF. Η άποψη της Πρωτοβουλίας Ανοικτού Κώδικα (*Open Source Initiative*), που διαφέρει στην φιλοσοφική της προσέγγιση, αναλύεται στον κατάλογο (*Open Source Initiative*, "Open Source licences") [181]. Μπορούμε να διαπιστώσουμε αποκλίσεις σε μερικές άδειες, σαν την Apple Public Source Licence Ver. 1.2, την οποία το FSF θεωρεί ως Μη-Ελεύθερη, λόγω της υποχρέωσης να δημοσιεύονται όλες οι αλλαγές (ακόμη κι' αν είναι ιδιωτικές), να ειδοποιείται η Apple για κάθε αναδιανομή, ή και της δυνατότητας μονόπλευρης ανάκλησης. Παρ' όλα αυτά, η πίεση που ασκήθηκε μέσω αυτής της κατάταξης, οδήγησε την Apple να δημοσιεύσει την έκδοση 2.0, τον Αύγουστο του 2003, την οποία το FSF θεώρησε, κατόπιν, ως Ελεύθερη.

Μπορούμε να κατατάξουμε τις Άδειες Ελεύθερου Λογισμικού σε δύο μεγάλες οικογένειες. Η πρώτη περιλαμβάνει άδειες που δεν επιβάλλουν ειδικές συνθήκες στη δεύτερη αναδιανομή (με άλλα λόγια, οι οποίες διευκρινίζουν μόνον ότι το λογισμικό μπορεί να αναδιανεμάται ή να τροποποιείται, αλλά χωρίς να επιβάλλουν ειδικούς όρους για να συμβεί αυτό, πράγμα που επιτρέπει, για παράδειγμα, σε κάποιον που έλαβε το πρόγραμμα, να μπορεί να το αναδιανείμει ως ιδιοταγές κλειστό λογισμικό): αυτές είναι οι άδειες στις οποίες θα αναφερόμαστε ως "Ανεκτικές ή Χαλαρές" Άδειες (*"permissive licences"*).

Η δεύτερη οικογένεια, που θα αποκαλούμε "Ισχυρές Άδειες" (ή και Άδειες copyleft), περιλαμβάνουν εκείνες που, σε συντονισμό με το στυλ της GPL του έργου GNU, επιβάλλουν όρους στην περίπτωση που κάποιος θέλει να αναδιανείμει το λογισμικό, οι οποίοι αποσκοπούν να διασφαλίσουν την συμμόρφωση με τους όρους της Αδείας μετά την πρώτη αναδιανομή. Ενώ η πρώτη ομάδα δίνει έμφαση στην Ελευθερία του ατομικού χρήστη που αποδέχεται το πρόγραμμα, να πράξει σχεδόν ό,τι επιθυμεί με αυτό (από την σκοπιά των όρων για τις μελλοντικές αναδιανομές), η δεύτερη ομάδα δίνει έμφαση στην ελευθερία οποιουδήποτε που θα μπορούσε, ενδεχομένως, να γίνει μία μέρα αποδέκτης ενός έργου που να προέρχεται από το πρόγραμμα, υποχρεώνοντας τις επόμενες τροποποιήσεις και τις αναδιανομές να σέβονται τους όρους της αρχικής Αδείας.

Η διαφορά ανάμεσα σε αυτούς τους δύο τύπους αδειάς ήταν (και παραμένει) ένα συζητήσιμο θέμα μέσα στην κοινότητα Ελεύθερου Λογισμικού. Εν πάσει περιπτώσει, θα πρέπει να θυμόμαστε ότι όλες αυτές είναι ελεύθερες άδειες.

3.2.2. Ανεκτικές (ή “Χαλαρές”) Άδειες

Οι *Ανεκτικές Άδειες*, ενίοτε γνωστές και ως “*Φιλελεύθερες*” (*liberal*) ή και ως “*Ελάχιστες*” (*minimal*) Άδειες, δεν επιβάλλουν, στην κυριολεξία, καμία υποχρέωση στον αποδέκτη του προγράμματος και, επιπλέον, του παραχωρούν την άδεια χρήσης, αναδιανομής και τροποποίησης. Από μία άποψη, αυτή η προσέγγιση μπορεί να θεωρηθεί ως μία εγγύηση για τον ύψιστο βαθμό ελευθερίας του αποδέκτη του προγράμματος. Αλλά από την άλλη, μπορεί και να ερμηνευθεί ως ένας μέγιστος βαθμός παραμέλησης, ως προς το κατά πόσο διασφαλίζεται ότι όταν κάποιος αποδέχεται ένα πρόγραμμα, αυτό το ίδιο πρόσωπο θα εγγυηθεί τις ίδιες ελευθερίες και όταν θα το αναδιανείμει, στη συνέχεια. Στην πράξη, αυτές οι Άδειες επιτρέπουν τυπικά σε ένα Λογισμικό, του οποίου ο συγγραφέας/ δημιουργός το κυκλοφορεί υπό μία Ανεκτική Άδεια, να μπορεί να αναδιανεμηθεί και με μία ιδιοταγή, κλειστή άδεια.

Ανάμεσα σε αυτού του τύπου τις άδειες, η άδεια BSD είναι η πλέον γνωστή, σε τέτοιο βαθμό που συχνά αναφερόμαστε, ακόμη και στις άλλες Ανεκτικές Άδειες, ως Άδειες *τύπου BSD*. Η Άδεια BSD (Berkeley Software Distribution) προήλθε από την δημοσίευση των διαφόρων εκδόσεων του Unix, που παρήχθησαν στο Πανεπιστήμιο της California, του Berkeley, στις ΗΠΑ. Η μόνη υποχρέωση που επιβάλλει, είναι να αναφέρονται οι αρχικοί συγγραφείς, ενώ επιτρέπει την αναδιανομή τόσο υπό μορφή binary, όσο και υπό μορφή πηγαίου κώδικα, χωρίς να επιβάλλει, σε καμία περίπτωση, καμία από τις δύο μορφές. Δίνει επίσης την άδεια να γίνονται οποιεσδήποτε αλλαγές και να ενσωματώνονται μέσα σε άλλα προγράμματα, χωρίς κανέναν περιορισμό.

Σημείωση

Ο όρος *copyleft*, όταν εφαρμόζεται σε μία Άδεια, χρησιμοποιείται κυρίως από το Ίδρυμα Ελεύθερου Λογισμικού (Free Software Foundation) για να αναφερθούν στις δικές τους Άδειες, και έχει παρόμοιες επιπτώσεις με εκείνες που αναφέρονται ως *Ισχυρές Άδειες*, με την έννοια που χρησιμοποιούμε σε αυτό το κείμενο.

Σημείωση

Μία από τις πρακτικές συνέπειες των αδειών τύπου BSD, ήταν η διάδοση των προδιαγραφών, αφού οι developers δε συναντούν εμπόδια στο φτιάξιμο προγραμμάτων συμβατών με μία αρχική υλοποίηση (implementation) αναφοράς, στα πλαίσια αυτής της αδειάς. Στην πραγματικότητα, αυτός είναι ένας από τους λόγους για την εξαιρετική και τάχιστα διάδοση των πρωτοκόλλων του Διαδικτύου και της προγραμματιστικής διεπαφής sockets-based, διότι οι περισσότεροι εμπορικοί developers πήραν τη δική τους υλοποίηση (implementation) από εκείνη του Πανεπιστημίου του Berkeley.

Οι Ανεκτικές Άδειες είναι αρκετά δημοφιλείς, και υπάρχει μία ολόκληρη οικογένεια με παρόμοια χαρακτηριστικά με το BSD: η X Window, η Tcl/Tk, η Apache, κλπ. Από ιστορική άποψη, αυτές οι άδειες εμφανίστηκαν διότι το ανάλογο λογισμικό είχε αναπτυχθεί από Πανεπιστήμια, στο πλαίσιο κάποιων ερευνητικών προγραμμάτων, που χρηματοδοτήθηκαν από τη κυβέρνηση των ΗΠΑ. Τα Πανεπιστήμια δεν πωλούσαν αυτά τα προγράμματα, με το σκεπτικό ότι είχαν ήδη πληρωθεί γι' αυτά από την Κυβέρνηση, και άρα, από τον φορολογούμενο, πράγμα που σήμαινε ότι οποιαδήποτε εταιρεία ή φυσικό πρόσωπο θα μπορούσε να τα χρησιμοποιήσει, χωρίς κανένα, σχεδόν, περιορισμό.

Όπως ήδη είπαμε, επάνω σε ένα πρόγραμμα που διενεμήθη με μία Ανεκτική Άδεια, μπορεί να φτιαχτεί άλλο ένα (στην πραγματικότητα, θα πρόκειται για μία νέα έκδοση), το οποίο μπορεί να είναι ιδιοταγές. Οι επικριτές των αδειών τύπου BSD, διαβλέπουν έναν κίνδυνο σε αυτό το χαρακτηριστικό, αφού δεν εγγυάται την ελευθερία των μελλουσών εκδόσεων του προγράμματος. Οι υποστηρικτές της, αντιθέτως, την θεωρούν ως την ύψιστη έκφραση ελευθερίας και αντιτείνουν ότι, σε τελευταία ανάλυση, μπορείς να πράξεις (σχεδόν) ο,τιδήποτε θέλεις με το λογισμικό.

Οι περισσότερες Ανεκτικές Άδειες αποτελούν μία λέξη-προς-λέξη αντιγραφή της αρχικής αδειάς του Berkeley, με τροποποιήσεις μόνον σε ό,τι αφορά την ιδιότητα του συγγραφέως (*authorship*). Σε μερικές περιπτώσεις,

όπως η άδεια του Apache project, περιλαμβάνεται μία πρόσθετη ρήτρα, που είναι η απαγόρευση της απόδοσης του ίδιου ονόματος στις επανεκδόσεις, όπως εκείνο της αρχικής έκδοσης. Όλες αυτές οι άδειες, συνήθως περιλαμβάνουν, σαν την BSD, την απαγόρευση χρήσης του ονόματος το κατόχου των δικαιωμάτων, για την προώθηση των παράγωγων προϊόντων.

Συγχρόνως, όλες οι άδειες, είτε τύπου BSD είτε όχι, περιλαμβάνουν έναν περιορισμό της εγγύησης, που είναι στην πραγματικότητα μία αποποίηση ευθύνης (*disclaimer*), εγγύησης που είναι απαραίτητη για την αποφυγή έγερσης νομικών απαιτήσεων για υπονοούμενες εγγυήσεις. Αν και αυτή η αποκλήρυξη στο Ελεύθερο Λογισμικό έχει καταστεί αντικείμενο ευρείας κριτικής, εν τούτοις, αποτελεί κοινή πρακτική με το ιδιοταγές λογισμικό, το οποίο, γενικώς, εγγυάται μόνο ότι το μέσο (*medium*) είναι σωστό και ότι το εν λόγω πρόγραμμα τρέχει.

Περίγραμμα Περίληψης της Άδειας BSD

Copyright © the owner. All rights reserved.

Επιτρέπεται η αναδιανομή και η χρήση, υπό μορφή πηγαίου κώδικα και υπό δυαδική μορφή, με ή χωρίς τροποποιήσεις, υπό την προϋπόθεση ότι πληρούνται οι παρακάτω συνθήκες:

1) Οι αναδιανομές του πηγαίου κώδικα θα πρέπει να αναφέρουν την ειδοποίηση περί του *copyright*, και να παραθέτουν έναν κατάλογο αυτών των όρων καθώς και της αποποίησης ευθύνης.

2) Οι αναδιανομές σε δυαδική μορφή θα πρέπει να αναπαραγάγουν την ειδοποίηση περί του *copyright* και να παραθέτουν κατάλογο αυτών των όρων καθώς και της αποποίησης, μέσα στην έγγραφη τεκμηρίωση.

3) Ούτε το όνομα του *ιδιοκτήτη*, ούτε τα ονόματα των ανθρώπων που συνεισέφεραν, μπορούν να χρησιμοποιηθούν, για να προβληθούν προϊόντα που προήλθαν από αυτό το λογισμικό, χωρίς την άδεια.

Αυτό το πρόγραμμα παρέχεται “ως έχει”, και αποκηρύσσονται όλες οι ρητές ή υπονοούμενες εγγυήσεις εμπορικότητας ή καταλληλότητας για έναν συγκεκριμένο σκοπό. Ο ιδιοκτήτης δεν μπορεί να θεωρηθεί υπεύθυνος, σε καμία περίπτωση, για βλάβες που μπορεί να προεκληθούν από τη χρήση του (περιλαμβανομένης της απώλειας δεδομένων, απώλειας κερδών, ή και της διακοπής επιχειρηματικής δραστηριότητας).

Στα επόμενα εδάφια θα περιγράψουμε, εν συντομία, μερικές *Ανεκτικές ή Φιλελεύθερες (Permissive) Άδειες*:

• Η Άδεια X Window, έκδοση 11 (X11) (<http://www.x.org/Downloads/terms.html>) [73].

Αυτή είναι η άδεια που χρησιμοποιείται για την διανομή του συστήματος X-Window, το πιο εκτεταμένα χρησιμοποιούμενο σύστημα διαχείρισης παραθύρων στον κόσμο του Unix, καθώς επίσης σε περιβάλλοντα GNU/Linux. Είναι πολύ παρόμοιο με την άδεια BSD, η οποία επιτρέπει αναδιανομή, χρήση και τροποποίηση, χωρίς ιδιαίτερους περιορισμούς. Μερικές φορές αποκαλείται και Άδεια MIT (με επικίνδυνη έλλειψη ακρίβειας, αφού το MIT έχει χρησιμοποιήσει άλλους τύπους Αδειών). Τα έργα που προέρχονται από το σύστημα X-Window, όπως για παράδειγμα το Xfree86, διανέμονται επίσης υπό αυτή την συγκεκριμένη Άδεια.

• Η Άδεια Zope Public 2.0 (<http://www.zope.org/Resources/ZPL>) [76].

Αυτή η Άδεια (ευρέως γνωστή ως ZPL) χρησιμοποιείται για την διανομή του Zope (έναν διακομιστή εφαρμογών) και για άλλα σχετικά προϊόντα. Είναι παρόμοια με το BSD, με το περίεργο χαρακτηριστικό ότι απαγορεύει ρητώς την χρήση των trademarks που έχουν κατατεθεί από την Εταιρεία Zope.

• Η Άδεια Apache.

Αυτή είναι η Άδεια υπό την οποία διανέμονται τα περισσότερα προϊόντα που παράγει το έργο Apache project. Είναι παρόμοια με την Άδεια BSD.

Υπάρχουν μερικά Ελεύθερα προγράμματα που δεν διανέμονται με καμία συγκεκριμένη Άδεια, αλλά για τα οποία ο συγγραφέας δηλώνει ξεκάθαρα ότι ανήκουν σε Δημόσια Χρήση (*public domain*). Το βασικό αποτέλεσμα αυτής της δήλωσης είναι ότι ο συγγραφέας αποποιείται όλων των δικαιωμάτων επί του προγράμματος, το οποίο, πλέον, μπορεί να τροποποιείται, να διανέμεται, να χρησιμοποιείται, κλπ, με οποιονδήποτε τρόπο. Πρακτικά μιλώντας, είναι πολύ παρόμοιο με το να κυκλοφορεί ένα πρόγραμμα με μία Άδεια τύπου BSD.

3.2.3. Ισχυρές Άδειες

Η Γενική Δημόσια Άδεια χρήσης GNU (GNU General Public Licence ή GNU GPL)

Η General Public Licence του GNU project (Free Software Foundation, 1991) [118] (περισσότερο γνωστή με τα αρχικά της, ως GPL), την οποία μπορείτε να βρείτε στο Παράρτημα C, είναι, μακράν, η πιο δημοφιλής και ευρέως γνωστή στον κόσμο του Ελεύθερου Λογισμικού. Δημιουργήθηκε αρχικά από το Ίδρυμα Ελεύθερου Λογισμικού (Free Software Foundation, που ήταν και ο βασικός συντελεστής για το GNU project), και είχε επινοηθεί εξ' αρχής να είναι η Άδεια για όλα τα λογισμικά που θα παρήγαγε το FSF.

Εν τούτοις, η χρήση της διαδόθηκε περαιτέρω και έγινε η Άδεια που χρησιμοποιείται περισσότερο (για παράδειγμα, πάνω από το 70% των projects που δημοσιεύονται στο Freshmeat, αδειοδοτούνται υπό τους όρους της GPL), ακόμη και από έργα-ναυαρχίδες του κόσμου του Ελεύθερου Λογισμικού, όπως ο Πυρήνας Linux.

Η Άδεια GPL είναι ενδιαφέρουσα από Νομικής πλευράς, διότι χρησιμοποιεί δημιουργικά τη νομοθεσία περί πνευματικών δικαιωμάτων και copyright, για να πετύχει πρακτικά το αντίθετο αποτέλεσμα από εκείνο που η Νομοθεσία επεδίωκε: αντί να περιορίζει τα δικαιώματα των χρηστών, τα εγγυάται. Γι' αυτό τον λόγο, η όλη διαδικασία αποκαλείται συχνά και copyleft (ένα λογοπαίγνιο, με την αντικατάσταση του συστατικού "right", της λέξης "copyright", με τον όρο "left"). Κάποιοι με αίσθηση του χιούμορ, επινόησαν ακόμη και το σλόγκαν "copyleft, all rights reversed" ("copyleft, όλα τα δικαιώματα αντεστραμμένα").

Βασικά, η άδεια GPL επιτρέπει την αναδιανομή σε δυαδική μορφή (εκτελέσιμο) αλλά και σε Πηγαίο Κώδικα, αν και στην περίπτωση διανομής υπό δυαδική μορφή (εκτελέσιμου αρχείου) είναι επίσης υποχρεωτική η πρόσβαση στον Πηγαίο Κώδικα. Επιτρέπει, εξ' άλλου, να γίνονται τροποποιήσεις, χωρίς καθόλου περιορισμούς. Σε κάθε περίπτωση, η διανομή Κώδικα που έχει αδειοδοτηθεί υπό τους όρους της GPL, που είναι όμως ενσωματωμένος με άλλον κώδικα (δηλ. συνδεδεμένα κομμάτια κώδικα), είναι δυνατή μόνο εάν αυτός διαθέτει μία συμβατή Άδεια. Αυτό το χαρακτηριστικό είναι γνωστό και ως "φαινόμενο ιογενούς μετάδοσης" της GPL (αν και πολλοί θεωρούν ότι αυτή η έκφραση μπορεί να εκληφθεί και ως έλλειψη σεβασμού), αφού από την στιγμή που ο κώδικας δημοσιεύθηκε υπό αυτούς τους συγκεκριμένους όρους, τότε αυτοί δε μπορούν ποτέ πλέον να αλλάξουν.

Σημείωση

Μία Άδεια είναι ασύμβατη με την GPL όταν περιορίζει οποιαδήποτε δικαιώματα που εγγυάται η GPL, είτε κατά τρόπο εμφανή, διατυπώνοντας αντιρρήσεις σε οποιεσδήποτε ρήτρες της GPL, είτε και έμμεσα, δια της επιβολής κάποιων νέων περιορισμών. Για παράδειγμα, η τρέχουσα έκδοση της Αδείας BSD είναι συμβατή, ενώ, αντιθέτως, η Άδεια Apache είναι ασύμβατη, διότι απαιτεί όλα τα διαφημιστικά υλικά να αναφέρουν ρητώς πως το συνδυασμένο έργο περιέχει κώδικα του καθενός μεμονωμένου κατόχου των δικαιωμάτων. Αυτό δεν σημαίνει ότι τα προγράμματα που διανέμονται και με τις 2 άδειες δεν μπορούν να χρησιμοποιούνται ταυτοχρόνως, ή ακόμη και να ενσωματώνονται. Σημαίνει μόνον ότι εκείνα τα προγράμματα που βρίσκονται ενσωματωμένα δεν μπορούν να διανέμονται, αφού είναι φύσει αδύνατον να συμμορφώνονται συγχρόνως με τους όρους αναδιανομής και των δύο.

Η Άδεια GPL έχει σχεδιαστεί για να εγγυάται την ελευθερία του Πηγαίου Κώδικα σε κάθε περίπτωση, αφού ένα πρόγραμμα που δημοσιεύεται και αδειοδοτείται υπό τους όρους της, δεν μπορεί ποτέ να μετατραπεί σε Ιδιοταγές. Επιπλέον, ούτε το ίδιο το πρόγραμμα αλλά ούτε και οι τροποποιήσεις του μπορούν να δημοσιευθούν υπό άλλη διαφορετική άδεια, εκτός από την GPL. Όπως ήδη τονίσαμε, οι υποστηρικτές των Αδειών τύπου BSD διαβλέπουν έναν περιορισμό της ελευθερίας σε αυτή τη συγκεκριμένη ρήτρα, ενώ οι οπαδοί της GPL πιστεύουν ότι πρόκειται για έναν τρόπο διασφάλισης ότι το Λογισμικό θα είναι πάντα Ελεύθερο. Ένας τρόπος να δούμε το θέμα αυτό, θα μπορούσε να είναι ότι η Άδεια GPL μεγιστοποιεί την ελευθερία των χρηστών, ενώ η Άδεια BSD μεγιστοποιεί την ελευθερία των παραγωγών/developers. Σε κάθε περίπτωση, θα πρέπει να σημειώσουμε ότι στη δεύτερη περίπτωση, αναφερόμαστε

γενικά στους παραγωγούς/ developers και όχι στους συγγραφείς, καθόσον πολλοί συγγραφείς θεωρούν ότι η Άδεια GPL είναι πιο κοντά στα δικά τους συμφέροντα, αφού υποχρεώνει τους ανταγωνιστές να δημοσιεύουν τις τροποποιήσεις τους (βελτιώσεις, διορθώσεις, κλπ.) όταν αναδιανέμουν το πρόγραμμά τους, ενώ στην περίπτωση της Άδειας τύπου BSD, κάτι τέτοιο δεν είναι απαραίτητο να συμβεί.

Όσον αφορά την φύση της Άδειας, που είναι αντίθετη με τα πνευματικά δικαιώματα (*copyright*), είναι διότι η φιλοσοφία της, όπως και εκείνη του Ιδρύματος Ελεύθερου Λογισμικού, υπαγορεύει ότι το Λογισμικό δεν πρέπει να έχει ιδιοκτήτες (Richard Stallman, "Why software should not have owners", 1998) [207]. Αν και αληθεύει ότι και το Λογισμικό που αδειοδοτείται υπό τους όρους της GPL, διαθέτει έναν συγγραφέα, που είναι εκείνος που επιτρέπει την εφαρμογή της νομοθεσίας περί πνευματικών δικαιωμάτων στην περίπτωση αυτή. Όμως, οι όροι υπό τους οποίους δημοσιεύεται, προσδίδουν μία τόσο ιδιαίτερη φύση στο Λογισμικό, που μπορούμε να θεωρήσουμε ότι η ιδιοκτησία ανήκει στο πρόσωπο που κατέχει το πρόγραμμα, παρά στο πρόσωπο που το δημιούργησε.

Οποσδήποτε, αυτή η Άδεια περιέχει και δηλώσεις αποποίησης ευθύνης (*disclaimers*), για να προστατεύει τους συγγραφείς. Παρομοίως, για να προστατευθεί η καλή φήμη των αρχικών συγγραφέων, οποιαδήποτε τροποποίηση ενός αρχείου του πηγαίου κώδικα, πρέπει να περιλαμβάνει και μία σημείωση που να φέρει την ημερομηνία και το όνομα του συγγραφέως της τροποποίησης. Η GPL λαμβάνει, επίσης, υπ' όψη της τα διπλώματα ευρεσιτεχνίας (*patents*) Λογισμικού, και απαιτεί, αν ο Κώδικας περιέχει πατενταρισμένους αλγορίθμους (όπως είπαμε, αυτό είναι κάτι κοινό και νόμιμο στις ΗΠΑ, αλλά θεωρείται, τρεχόντως, αντικανονικό στην Ευρώπη), τότε είτε θα πρέπει να παραχωρείται μία δωρεάν Άδεια χρήσης του διπλώματος ευρεσιτεχνίας (*patent*), είτε δεν θα μπορεί να διανέμεται υπό την GPL.

Η πιο πρόσφατη έκδοση της Άδειας GPL, δηλαδή η δεύτερη, δημοσιεύθηκε το 1991 (κατά τη στιγμή που γράφεται το παρόν, υπάρχει, ήδη, μία 3^η έκδοση, σε προχωρημένο στάδιο προετοιμασίας). Ειδικά έχοντας κατά νου τις μελλοντικές εκδόσεις, η Άδεια συστήνει την υπαγωγή υπό τους όρους της 2^{ης} έκδοσης, ή των οποιωνδήποτε επόμενων που θα δημοσιευθούν από το Free Software Foundation, πράγμα που είναι ακριβώς εκείνο που πολλοί συγγραφείς πράττουν. Άλλοι, πάντως, περιλαμβανομένου, ειδικά, του Linus Torvalds (του δημιουργού του πυρήνα Linux), διανέμουν το λογισμικό τους υπό τους όρους της 2^{ης} έκδοσης της GPL, σε μία προσπάθεια να πάρει αποστάσεις από πιθανές μελλοντικές εξελίξεις του Free Software Foundation.

Η 3^η έκδοση της GPL (<http://gplv3.fsf.org>) [115] προτίθεται να επιφέρει μία επικαιροποίηση στο τρέχον σκηνικό των λογισμικών, κυρίως όσον αφορά ορισμένα θέματα όπως τα διπλώματα ευρεσιτεχνίας (*patents*), τα DRM (Διαχείριση Ψηφιακών Δικαιωμάτων/*Digital Rights Management*) και άλλους περιορισμούς στην ελευθερία του λογισμικού.

Για παράδειγμα, το προσχέδιο που είναι, προς το παρόν, διαθέσιμο, κατά τη στιγμή που γράφεται το παρόν (Μάιος 2007), δεν επιτρέπει σε έναν κατασκευαστή hardware να απαγορεύει τη χρήση κάποιων αρθρωμάτων λογισμικού (*software modules*), εάν δεν διαθέτουν μία ψηφιακή υπογραφή που να πιστοποιεί έναν συγκεκριμένο συγγραφέα. Ένα παράδειγμα αυτής της πρακτικής, απαντάται στην περίπτωση των ψηφιακών συσκευών καταγραφής TiVo, που παρέχουν τον πηγαίο κώδικα για όλα τα λογισμικά τους (που αδειοδοτούνται υπό την GPLv2), ενώ συγχρόνως δεν επιτρέπουν τροποποιήσεις του κώδικα που προορίζεται για χρήση στο υλικό αυτό (*hardware*)⁴.

⁽⁴⁾ Για την περίπτωση αυτή η περίπτωση προτάθηκε ακόμη και η χρήση της λέξης "tivoisation" (TiVo-ποίηση), για άλλες, παρόμοιες περιπτώσεις, που παρατηρήθηκαν.

Η Άδεια δεν επιτρέπει ούτε να τρέξει βεβαιωμένα ένα λογισμικό, σε ένα προκαθορισμένο

περιβάλλον, όπως συμβαίνει στην περίπτωση απαγόρευσης χρήσης μη πιστοποιημένων πυρήνων, σε διανομές που το κρίνουν σκόπιμο για λόγους ασφαλείας.

Σημείωση

Υπάρχουν διάφορα σημεία της Αδείας GPLv3 που προκάλεσαν κάποιες αντιδράσεις.

Μία από τις ομάδες που εκφράζουν αντίρρηση, είναι η ομάδα των developers του πυρήνα Linux (στην οποία ανήκει και ο ίδιος ο Linus Torvalds). Υποστηρίζουν ότι η απαίτηση για χρήση τμημάτων υπογεγραμμένου λογισμικού, επιτρέπει την παραχώρηση ορισμένων χαρακτηριστικών ασφαλείας που δεν θα ήταν, διαφορετικά, δυνατή, ενώ συγχρόνως, η ρητή τους απαγόρευση θα επεξέτεινε την Άδεια και στο πεδίο του hardware.

Επιπλέον, ο περιορισμός που υιοθετήθηκε από τον μηχανισμό υπογραφών, θα ίσχυε μόνο για τις πλατφόρμες hardware και software που σχεδιάστηκαν κατ' αυτόν τον τρόπο, εννοώντας ότι το λογισμικό θα μπορούσε να τροποποιηθεί για χρήση σε διαφορετικό hardware. Σχετικά με αυτό το σημείο, το FSF είναι υπέρ της χρήσης μηχανισμών υπογραφής που να συμβουλεύουν κατά της χρήσης των μη υπογεγραμμένων components, για λόγους ασφαλείας, αλλά πιστεύει ότι η μη απαγόρευση αυτών των μηχανισμών υπογραφής που προλαμβάνουν την χρήση ανυπόγραφων components, θα μπορούσε να οδηγήσει στη δημιουργία καταστάσεων όπου να μην υπάρχουν πλατφόρμες hardware ή software πάνω στις οποίες να τρέξουν οι τροποποιήσεις του λογισμικού. Αυτό θα σήμαινε ότι η ελευθερία του Ελεύθερου Λογισμικού θα καταντούσε εντελώς περιορισμένη, στις περιπτώσεις που θα αφορούν την χρήση τροποποιημένου κώδικα

Η GNU Ελάσσων Γενική Δημόσια Άδεια (Lesser General Public Licence, GNU LGPL)

Η GNU Ελάσσων Γενική Δημόσια Άδεια, (Free Software Foundation, GNU LGPL, version 2.1, Φεβρουάριος 1999) [119], στην οποία αναφερόμαστε συνήθως με τα αρχικά της- LGP - είναι μία άλλη άδεια του Ιδρύματος Ελεύθερου Λογισμικού. Σχεδιάστηκε αρχικά για να χρησιμοποιηθεί σε Βιβλιοθήκες (το γράμμα “L”, σήμαινε αρχικά “library”), και τροποποιήθηκε πρόσφατα, ώστε να θεωρείται η μικρότερη (ελάσσων) αδελφή της GPL.

Η LGPL επιτρέπει σε Ελεύθερα προγράμματα να χρησιμοποιούνται με Ιδιοταγές Λογισμικό. Το πρόγραμμα αυτό καθ' εαυτό, αναδιανέμεται σαν να ήταν υπό την Άδεια GPL, αλλά επιτρέπεται η ενσωμάτωσή του με οποιαδήποτε άλλα πακέτα λογισμικού χωρίς απολύτως κανέναν περιορισμό.

Όπως βλέπουμε, ενώ η Άδεια επινοήθηκε αρχικά για χρήση σε βιβλιοθήκες, για να προωθηθεί η χρήση τους και η ανάπτυξή τους, χωρίς τα προβλήματα ενσωμάτωσης που προέκυπταν από την GPL. Εν πάσει περιπτώσει, όταν έγινε κατανοητό ότι το επιδιωκόμενο αποτέλεσμα, δηλ. το να καταστούν δημοφιλείς οι Ελεύθερες βιβλιοθήκες, δεν αντισταθμίζονταν με αντίστοιχη δημιουργία Ελεύθερων προγραμμάτων, τότε το Free Software Foundation αποφάσισε να αλλάξει το “βιβλιοθήκη” σε “ελάσσονα” και τοποθετήθηκε ενάντια στη χρήση της, πλην πολύ περιστασιακών και ιδιαίτερων περιστάσεων. Στη σημερινή εποχή, υπάρχουν πολλά προγράμματα που δεν είναι βιβλιοθήκες, αλλά που έχουν αδειοδοτηθεί υπό την LGPL. Για παράδειγμα, ο περιηγητής Διαδικτύου Mozilla , ή η σουίτα γραφείου OpenOffice.org, είναι κι' αυτά αδειοδοτημένα, μεταξύ άλλων, με την LGPL.

Σημείωση

Όπως ισχύει και για την GPL, η τελευταία δημοσιευθείσα έκδοση της LGPL είναι η 2^η, παρά το ότι υπάρχει ήδη ένα προσχέδιο και μίας 3^{ης} έκδοσης (<http://gplv3.fsf.org/pipermail/info-gplv3/2006-July/000008.html>) [116]. Αυτή η νέα έκδοση είναι βραχύτερη από την προηγούμενη, αφού παραπέμπει για όλο το κείμενό της στην GPLv3 και απλώς αναδεικνύει τις διαφορές.

Άλλες Ισχυρές Άδειες

Άλλες Άδειες που αξίζουν μίας αναφοράς:

- **Η Άδεια Sleepycat** (www.sleepycat.com/download/oslicense.html) [59].

Αυτή είναι η Άδεια υπό την οποία η εταιρεία Sleepycat (<http://www.sleepycat.com/>) [60] διανέμει τα προγράμματά της (ανάμεσα στα οποία θα μπορούσαμε να αναφέρουμε το πολύ γνωστό Berkeley DB). Επιβάλλει κάποιους όρους, οποτεδήποτε το πρόγραμμα, ή τα εξ' αυτού παράγωγα έργα,

αναδιανέμονται. ειδικότερα, επιβάλλει να προσφέρεται ο Πηγαίος Κώδικας (περιλαμβάνοντας και τις τροποποιήσεις, στην περίπτωση ενός παράγωγου έργου) αλλά και στην αναδιανομή να υποχρεώνει τον αποδέκτη στους ίδιους όρους. Αν και βραχύτερη από την GNU GPL, είναι παρόμοια σε πολλά από τα αποτελέσματά της.

• **Η Άδεια eCos License 2.0** (<http://www.gnu.org/licenses/ecos-license.html>) [25].

Αυτή είναι η Άδεια υπό την οποία διανέμεται το eCos (<http://sources.redhat.com/ecos/>) [24], ένα λειτουργικό σύστημα πραγματικού χρόνου (*real-time operating system*). Πρόκειται για μία τροποποίηση της GNU GPL, που δεν θεωρεί ότι, σε περίπτωση αναδιανομής, ο κώδικας που συνδέεται με τα προγράμματα που προστατεύει, υπόκειται στις ρήτρες της GNU GPL. Από αυτή την άποψη, τα αποτελέσματά της είναι παρόμοια με εκείνα της GNU LGPL.

• **Γενική Δημόσια Άδεια Affero** (<http://www.affero.org/oagpl.html>) [78].

Πρόκειται για μία ενδιαφέρουσα τροποποίηση της GNU GPL, που εστιάζει στην περίπτωση των προγραμμάτων που προσφέρουν υπηρεσίες μέσω του Διαδικτύου, ή γενικότερα, μέσω δικτύων υπολογιστών. Αυτό το είδος προγραμμάτων αντιπροσωπεύει ένα πρόβλημα, από τη σκοπιά των Ισχυρών Αδειών. Από τη στιγμή που η χρήση του προγράμματος δεν προϋποθέτει να το έχει προμηθευθεί κάποιος από μία αναδιανομή, ακόμη και αν κυκλοφορεί, για παράδειγμα, υπό την άδεια GNU GPL, μπορεί κάποιος να το τροποποιήσει και να προσφέρει μία υπηρεσία στο Web μέσω της χρήσης του προγράμματος αυτού, χωρίς να το αναδιανέμει κατά κανέναν τρόπο, και άρα, χωρίς να είναι υποχρεωμένος, να αναδιανέμει τον Κώδικα. Η Άδεια Affero GPL προβλέπει μία ρήτρα με την υποχρέωση ότι αν το πρόγραμμα διαθέτει ένα τρόπο να παρέχει πρόσβαση στον Πηγαίο Κώδικα διά μέσω του web, προς οποιονδήποτε το χρησιμοποιεί, αυτό το χαρακτηριστικό δεν μπορεί να απενεργοποιηθεί. Αυτό σημαίνει ότι αν ο αρχικός συγγραφέας περιλάβει αυτή την δυνατότητα μέσα στον Πηγαίο Κώδικα, οποιοσδήποτε μπορεί πλέον να το έχει, πέραν του ότι επιπλέον η αναδιανομή υπόκειται στις συνθήκες της Αδείας. Το Ίδρυμα Ελεύθερου Λογισμικού εξετάζει να συμπεριλάβει παρόμοιες προβλέψεις στην έκδοση 3 της GNU GPL.

• **Η Δημόσια Άδεια IBM 1.0**

(<http://oss.software.ibm.com/developerworks/opensource/license10.html>) [40].

Πρόκειται για μία Άδεια που επιτρέπει στα παράγωγα έργα να αναδιανέμονται υπό δυαδική μορφή, μόνον αν (μεταξύ διαφόρων άλλων προϋποθέσεων) προβλέπει έναν μηχανισμό που να επιτρέπει στον αποδέκτη του προγράμματος να λαμβάνει τον Πηγαίο Κώδικα. Η αναδιανομή του Πηγαίου Κώδικα πρέπει να γίνεται υπό τους όρους της ίδιας Αδείας. Αυτή η Άδεια είναι ενδιαφέρουσα για τον πρόσθετο λόγο ότι υποχρεώνει το άτομο που αναδιανέμει ένα τροποποιημένο πρόγραμμα, να αδειοδοτεί αυτομάτως και δωρεάν οποιαδήποτε διπλώματα ευρεσιτεχνίας που αφορούν αυτές τις τροποποιήσεις, που αποτελούν ιδιοκτησία του αναδιανομέα, προς το ενδιαφερόμενο μέρος που αποδέχεται το πρόγραμμα.

• **Η Δημόσια Άδεια Mozilla 1.1** (<http://www.mozilla.org/MPL/MPL-1.1.html>) [49].

Αυτό είναι ένα παράδειγμα Ελεύθερης Αδείας, που σχεδιάστηκε από μία Εταιρεία. Αποτελεί μία εξέλιξη της πρωταρχικής Ελεύθερης Αδείας που διέθετε ο Netscape Navigator, που ήταν πολύ σημαντική για την εποχή εκείνη, διότι ήταν η πρώτη φορά που μία πολύ γνωστή εταιρεία αποφάσισε να διαθέσει ένα πρόγραμμα υπό μία δική της Ελεύθερη Άδεια.

3.2.4. Διανομή κάτω από περισσότερες Αδειες

Μέχρι τώρα, προϋποθέταμε ότι ένα πρόγραμμα διατίθεται υπό μία και μοναδική Άδεια, η οποία και περιγράφει τους όρους για την χρήση και την αναδιανομή του. Ένας συγγραφέας μπορεί, πάντως, να διανέμει τα έργα του και υπό πολλές διαφορετικές Αδειες. Για να μπορέσουμε να το καταλάβουμε

αυτό, πρέπει να θυμόμαστε ότι η κάθε δημοσίευση συνιστά και ένα νέο έργο και ότι μπορεί να διανέμονται διαφορετικές εκδόσεις, με τη μόνη διαφορά να έγκειται στον τύπο της Αδείας τους. Όπως θα δούμε, τις περισσότερες φορές αυτό σημαίνει, επίσης, ότι ο χρήστης, αναλόγως του τι ακριβώς θέλει να κάνει με το λογισμικό, πρέπει να τηρεί τους όρους της μίας ή της άλλης Αδείας.

Ένα από τα καλύτερα παραδείγματα μίας τέτοιας διπλής Αδείας είναι εκείνο της βιβλιοθήκης Qt, πάνω στην οποία στηρίζεται το περιβάλλον εργασίας KDE. Η εταιρεία Trolltech, με έδρα την Νορβηγία, δένει με την Qt με μία ιδιοταγή, κλειστή άδεια, παρά το ότι έστρωσε τον δρόμο για προγράμματα που δεν την χρησιμοποιούσαν για πραγματοποίηση κερδών. Γι' αυτό το λόγο, αλλά και λόγω των τεχνικών της χαρακτηριστικών, απετέλεσε την επιλογή του KDE project, κατά τα μέσα της δεκαετίας του '90. Αυτό δημιούργησε μία έντονη διχογνωμία με το Free Software Foundation, διότι το KDE έπαυε, έτσι, να συνιστά ένα εντελώς Ελεύθερο Λογισμικό, καθώς αποκτούσε εξάρτηση από μία ιδιοταγή βιβλιοθήκη. Μετά από μία εκτεταμένη αντιπαράθεση απόψεων (κατά τη διάρκεια της οποίας εμφανίστηκε το GNOME, ως ελεύθερος ανταγωνιστής του KDE, για το περιβάλλον επιφάνειας εργασίας των desktops), η Trolltech αποφάσισε να κλίνει υπέρ του συστήματος διπλής αδειοδότησης για το προϊόν αιχμής που διέθετε: τα προγράμματα υπό την GPL θα μπορούσαν να κάνουν χρήση της έκδοσης Qt GPL, ενώ εάν και όπου υπήρχε πρόθεση ενσωμάτωσης με προγράμματα που είχαν άδειες ασύμβατες με την GPL (όπως πχ. ιδιοταγείς κλειστές άδειες), τότε θα έπρεπε να αγοράσει κανείς μία ειδική άδεια από την εταιρεία. Αυτή η λύση ικανοποίησε όλες τις πλευρές, και στις ημέρες μας, το KDE θεωρείται, πλέον, ως Ελεύθερο Λογισμικό.

Άλλα πασίγνωστα παραδείγματα διπλής αδείας είναι το StarOffice και το OpenOffice.org, ή ο Netscape Communicator και το Mozilla. Και στις 2 περιπτώσεις, το πρώτο προϊόν είναι ιδιοταγές, ενώ το δεύτερο είναι μία Ελεύθερη έκδοση (σε γενικές γραμμές, υπό τους όρους διαφόρων ελευθέρων αδειών). Παρά το ότι, αρχικά, τα Ελεύθερα free projects αποτελούσαν περιορισμένες εκδόσεις των αντίστοιχων ιδιοταγών αδελφών τους, συν τω χρόνω, ακολούθησαν την δική τους πορεία, με αποτέλεσμα, στην εποχή μας, να έχουν ένα αρκετά υψηλό επίπεδο ανεξαρτησίας.

3.2.5. Τεκμηρίωση του προγράμματος

Το σύνολο της γραπτής τεκμηρίωσης που συνοδεύει ένα πρόγραμμα, συνιστά ένα αναπόσπαστο τμήμα του, όπως εξ' άλλου ισχύει και για τα σχόλια στον Πηγαίο Κώδικα, όπως αναγνωρίστηκε, για παράδειγμα, από την Ισπανική νομοθεσία σχετικά με την Πνευματική Ιδιοκτησία. Με δεδομένο αυτό το επίπεδο ενσωμάτωσης, θα ακουγόταν λογικό να ισχύουν οι ίδιες ελευθερίες και για την τεκμηρίωση και, γενικώς, τα πράγματα να εξελίσσονται όπως για το ίδιο το λογισμικό: κάθε τροποποίηση που γίνεται σε ένα πρόγραμμα, επιφέρει και την ανάγκη για μία ταυτόχρονη και αντίστοιχη αλλαγή και της τεκμηρίωσής του.

Το μεγαλύτερο τμήμα της τεκμηρίωσης τείνει να είναι κωδικοποιημένο υπό μορφή αμορφοποιητών αρχείων κειμένου, αφού το ζητούμενο είναι να γίνει όσο πιο διαθέσιμη γίνεται, με ένα περιβάλλον ελαχίστων εργαλείων (*minimum tools environment*), και (στη περίπτωση των Ελευθέρων προγραμμάτων) να περιλαμβάνει μία μικρή εισαγωγή για το πρόγραμμα (αρχείο Readme), οδηγίες εγκατάστασης (αρχείο Install), μερικά στοιχεία ιστορίας γύρω από την εξέλιξη και το μέλλον του προγράμματος (αρχεία ChangeLog και ToDo), τους συγγραφείς και το copyright (αρχεία Authors και Copyright ή Copying), καθώς και τις οδηγίες χρήσεως. Όλα αυτά, με την εξαίρεση των συγγραφέων και τα σχετικά περί copyright, πρέπει να είναι ελεύθερα τροποποιήσιμα, καθώς το πρόγραμμα εξελίσσεται. Στο κομμάτι με τους συγγραφείς, πρέπει, απλώς, να προσθέσουμε ονόματα και αναγνώριση συνεισφοράς (credits), χωρίς να απαλείψουμε τίποτε, ενώ το κομμάτι περί copyright θα πρέπει να τροποποιηθεί μόνον αν οι συνθήκες το επιτρέπουν.

Οι οδηγίες χρήσεως είναι, κανονικά, κωδικοποιημένες σε πιο σύνθετες μορφές αρχείων, από τη στιγμή που πρόκειται, συνήθως, για μεγαλύτερα και πλουσιότερα έγγραφα. Η φύση του Ελεύθερου Λογισμικού απαιτεί να μπορεί αυτή η τεκμηρίωση να είναι εύκολα τροποποιήσιμη. Αυτό, με τη σειρά του, οδηγεί στην ανάγκη χρήσης των λεγόμενων Διαφανών Προτύπων (transparent formats), δηλαδή με γνωστές προδιαγραφές και σε θέση να είναι επεξεργάσιμες από Ελεύθερα εργαλεία: έτσι, εκτός από το σκέτο και καθαρό κείμενο, μπορεί, για παράδειγμα, να χρησιμοποιείται η μορφή αρχείου για τους οδηγούς χρήσης (manual pages) του Unix, το TexInfo, το LaTeX ή το DocBook, χωρίς να αποκλείεται επίσης και η δυνατότητα διανομής του τελικού αποτελέσματος, με μία διαμόρφωση των εγγράφων σε πιο κατάλληλες μορφές για ανάγνωση επί της οθόνης ή για εκτύπωση, όπως η HTML, το PDF ή η μορφή RTF (συνήθως πρόκειται για πιο Αδιαφανή Πρότυπα/Opaque formats).

Η έγγραφη τεκμηρίωση του προγράμματος προετοιμάζεται, εν πάσει περιπτώσει, από τρίτους, οι οποίοι δεν είχαν ανάμειξη στην ανάπτυξη του προγράμματος. Ορισμένες φορές η τεκμηρίωση μπορεί να είναι και διδακτικής φύσεως, για να διευκολύνει την εγκατάσταση και την χρήση ενός συγκεκριμένου προγράμματος (αρχείο HOWTO). Άλλες φορές, πρόκειται για πιο εκτεταμένη τεκμηρίωση, που καλύπτει περισσότερα του ενός προγράμματα καθώς και τη διαλειτουργικότητά τους, που συγκρίνει λύσεις, κλπ., είτε υπό μορφή ενός tutorial, είτε υπό μορφή ενός εγχειριδίου αναφοράς. Μερικές φορές πρόκειται για μία απλή περίληψη των συχνότερων ερωτήσεων που υποβάλλουν οι χρήστες μαζί με τις απαντήσεις (αρχείο FAQ). Ένα αξιόλογο παράδειγμα είναι το project για την τεκμηρίωση του Linux (<http://www.tldp.org>) [44]. Σε αυτή την κατηγορία θα μπορούσαμε να συμπεριλάβουμε κάποια άλλα τεχνικά έγγραφα, που δεν έχουν απαραίτητα σχέση με προγράμματα: είτε πρόκειται για οδηγίες καλωδίωσης ενός τοπικού δικτύου, για την κατασκευή ενός ηλιακού φούρνου, για την επισκευή μιας μηχανής ή για την επιλογή προμηθευτή εργαλείων.

Αυτού του είδους τα έγγραφα είναι κάτι ανάμεσα σε ένα τυπικό βιβλιário οδηγιών ενός λογισμικού και σε πολύ τεχνικά και πρακτικής αξίας άρθρα ή βιβλία. Χωρίς να θίγονται οι ελευθερίες ανάγνωσης, αντιγραφής, τροποποίησης και αναδιανομής, ο συγγραφέας μπορεί να επιθυμεί να εκφράσει απόψεις που δεν θα ήθελε να παραποιηθούν, ή τουλάχιστον δεν θα ήθελε να σημειωθούν παρανοήσεις που να του αποδοθούν, είτε μπορεί να επιθυμεί να διατηρήσει κάποιες παραγράφους, όπως την έκφραση ευχαριστιών και αναγνώρισης, είτε να καθιστά υποχρεωτική την τροποποίηση κάποιων άλλων, όπως τον τίτλο. Παρά το ότι αυτές οι ανησυχίες μπορεί να προκύψουν εξ' ίσου και με τα λογισμικά, εν τούτοις δεν έχουν έως τώρα εκφρασθεί με την ίδια σφοδρότητα στον κόσμο του Ελεύθερου Λογισμικού, όπως συνέβη, αντιθέτως, στον κόσμο της ελεύθερης συγγραφής έργων (free documentation).

3.3. Περίληψη

Σε αυτό το κεφάλαιο εξετάσαμε τις νομικές πλευρές που διέπουν ή που επηρεάζουν τον κόσμο του Ελεύθερου Λογισμικού. Αποτελούν μέρος μίας ολόκληρης νομοθεσίας περί πνευματικής ή περί βιομηχανικής/ επιχειρηματικής (industrial) ιδιοκτησίας που επινοήθηκε, επί της αρχής, για να κεντρίσει τη δημιουργικότητα, μέσω της επιβράβευσης των αρχικών δημιουργών για μία προκαθορισμένη περίοδο. Από όλες τις διάφορες υποκατηγορίες, το λεγόμενο copyright είναι εκείνο που επηρεάζει περισσότερο το Ελεύθερο Λογισμικό και, εφ' όσον εφαρμοσθεί κατάλληλα, μπορεί να χρησιμοποιηθεί για να εγγυηθεί την ύπαρξη του Ελεύθερου Λογισμικού, μέσα από την μορφή των Ελεύθερων Αδειών.

Εξετάσαμε τη σημασία που έχουν οι Άδειες στον κόσμο του Ελεύθερου Λογισμικού και παρουσιάσαμε, επίσης, την τεράστια ποικιλία των υπαρχουσών Αδειών, τις βάσεις πάνω στις οποίες

στηρίζονται, τις επιδράσεις τους, τα πλεονεκτήματα και τα μειονεκτήματά τους.

Σαφώς, μπορούμε να πούμε ότι η GPL προσπαθεί να μεγιστοποιήσει την ελευθερία των χρηστών λογισμικού, είτε αυτοί πήραν το Ελεύθερο Λογισμικό απ' ευθείας από το δημιουργό του, είτε όχι, ενώ οι Άδειες τύπου BSD προσπαθούν να μεγιστοποιήσουν την ελευθερία εκείνων που επιθυμούν να επιφέρουν τροποποιήσεις ή εκείνων που αναδιανέμουν το πρόγραμμα.

Με βάση αυτά που είδαμε σε αυτό κεφάλαιο, μπορούμε να συμπεράνουμε ότι είναι πολύ σημαντικό να αποφασίσουμε από νωρίς το είδος της Αδείας που θα έχει ένα project, καθώς και να είμαστε πλήρως ενήμεροι για τα πλεονεκτήματα και τα μειονεκτήματά της, αφού μία τροποποίηση που επιφέρουμε σε δεύτερη φάση, μπορεί να αποδειχθεί εξαιρετικά δύσκολη υπόθεση, ειδικά εάν υπάρχει ένας μεγάλος αριθμός εξωτερικών συνεργατών που συνεισέφεραν.

Συμπερασματικά, θα θέλαμε να τονίσουμε ότι το Ελεύθερο Λογισμικό και το ιδιοταγές (κλειστό) λογισμικό, διαφέρουν αποκλειστικά και μόνον ως προς τους όρους της Αδείας υπό την οποία τα προγράμματα αυτά κυκλοφορούν. Στα επόμενα κεφάλαια, πάντως, θα δούμε ότι αυτή η καθαρά νομική διαφορά, μπορεί να επηρεάσει ή όχι τον τρόπο με τον οποίο αναπτύσσεται το λογισμικό, οδηγώντας στην γέννηση ενός νέου μοντέλου ανάπτυξης, που μπορεί να διαφέρει από τις μεθόδους "συμβατικής" ανάπτυξης που χρησιμοποιεί η βιομηχανία λογισμικού, σε μεγαλύτερο ή μικρότερο βαθμό, ανάλογα με την κάθε περίπτωση.

4. Οι Developers/ προγραμματιστές και τά κίνητρά τους

"Το να είσαι hacker, έχει πολύ πλάκα, αλλά είναι ένα είδος πλάκας για την οποία χρειάζεται πολλή προσπάθεια. Και για να καταβληθεί η προσπάθεια, απαιτούνται κίνητρα. Οι επιτυχημένοι αθλητές αντλούν τα δικά τους κίνητρα από κάποιες μορφές φυσική απόλαυση με το να καταφέρνουν τέτοιες επιδόσεις με το σώμα τους, με το να σπρώχνουν τους εαυτούς τους πέραν των σωματικών τους ορίων. Κατά παρόμοιο τρόπο, για να είσαι hacker, πρέπει να να νοιώθεις μία κάποια βασική συγκίνηση (thrill) με το να λύνεις προβλήματα, οξύνοντας τις ικανότητές σου και εξασκώντας την εξυπνάδα σου."
Eric Steven Raymond, "How to become a hacker"

4.1. Εισαγωγή

Ο εν μέρει ανώνυμος και αποκεντρωμένος (*distributed*) τρόπος με τον οποίο αναπτύσσεται το Ελεύθερο Λογισμικό, σήμαινε ότι, για πολλά χρόνια, όλοι οι ανθρώπινοι πόροι που εμπλεκόταν σε αυτό, παρέμεναν ευρέως άγνωστοι. Το αποτέλεσμα αυτής της άγνοιας ήταν η μυθοποίηση, τουλάχιστον σε κάποιο βαθμό, όλου του κόσμου του Ελεύθερου Λογισμικού και της ζωής εκείνων που βρίσκονται από πίσω, που βασιζόταν σε λίγο-πολύ διαδεδομένα στερεότυπα για την κουλτούρα των hacker και των υπολογιστών. Τα τελευταία λίγα χρόνια, η επιστημονική κοινότητα κατέβαλλε μία τεράστια προσπάθεια να γνωρίσει καλύτερα τους ανθρώπους που συμμετέχουν στα projects Ελεύθερου Λογισμικού, τα κίνητρά τους, το εκπαιδευτικό τους υπόβαθρο, και οποιεσδήποτε άλλες εν δυνάμει σημαντικές πλευρές. Από καθαρά πραγματιστικής απόψεως, το να γνωρίζουμε ποιός εμπλέκεται σε αυτού του είδους τα projects και το γιατί, μπορεί να αποδειχθεί εξαιρετικά χρήσιμο όταν πρόκειται να δημιουργήσουμε Ελεύθερο Λογισμικό. Μερικοί επιστήμονες, κυρίως Οικονομολόγοι, Ψυχολόγοι και Κοινωνιολόγοι, θέλησαν να πάνε και παραπέρα και να δούνε μέσα σε αυτή την Κοινότητα τους σπόρους μελλοντικών εικονικών (*virtual*) κοινοτήτων, με τους δικούς τους κανόνες και τη δική τους ιεραρχία, σε πολλές περιπτώσεις εντελώς διαφορετικές από εκείνες που γνωρίζουμε μέσα στην "παραδοσιακή" κοινωνία. Ένα από τα πιο σημαντικά μυστήρια προς εξιχνίαση, ήταν να μάθουμε ποια ήταν τα κίνητρα αυτών των προγραμματιστών λογισμικού για να συμμετάσχουν σε μία κοινότητα αυτής της φύσεως, με δεδομένο ότι τα οικονομικά πλεονεκτήματα, τουλάχιστον τα άμεσα, είναι ουσιαστικά ανύπαρκτα, ενώ και τα έμμεσα είναι δύσκολο να μετρηθούν και να ποσοτικοποιηθούν.

4.2. Ποιοι είναι οι developers/ Προγραμματιστές;

Αυτό το κεφάλαιο στοχεύει να προσφέρει μία συνολική εικόνα των ανθρώπων που ξοδεύουν τον χρόνο τους και τις ικανότητές τους, για να συμμετέχουν σε projects Ελεύθερου Λογισμικού. Τα δεδομένα που δείχνουμε προέρχονται κυρίως από επιστημονικές έρευνες των τελευταίων 10 ετών, από τις πιο σημαντικές, αν και σε καμία περίπτωση οι μόνες, περιλαμβάνοντας το *"Free/libre and open source software. Survey and study, part IV: "Survey of developers", 2002 [126]*, και το *"Who is doing it? Knowing more about libre software developers", 2001 [197]*.

Οι προγραμματιστές και developers λογισμικού είναι συνήθως νέα άτομα. Η μέση ηλικία είναι περί τα 27. Η διακύμανση ηλικίας είναι σημαντική, με κυρίαρχη την ομάδα από 21 έως και 24 ετών, ενώ η περισσότερο εμφανιζόμενη ηλικία είναι τα 23 έτη. Είναι σημαντικό να παρατηρήσουμε ότι η ηλικία κατά την οποία έγινε η προσέγγιση του Κινήματος του Ελεύθερου Λογισμικού κορυφώνεται ανάμεσα στα 18 και τα 25 και είναι ιδιαίτερα έντονο ανάμεσα στις ηλικίες των 21 και των 23 ετών, πράγμα που συμπίπτει με την ηλικία εισόδου στο Πανεπιστήμιο και της φοιτητικής ζωής.

Αυτά τα στοιχεία έρχονται σε αντίθεση με την άποψη ότι το Ελεύθερο Λογισμικό είναι, επί το πλείστον, μία υπόθεση για εφήβους, αν και είναι προφανής μία ανάμειξη και των εφήβων (περίπου το 20% των developers είναι λιγότερο από 20 ετών). Μετά βεβαιότητας, εκείνο που διαπιστώνουμε είναι ότι το μεγαλύτερο μέρος των developers (60%) βρίσκονται γύρω στα 20 τους, με τους κάτω των 20 και τους άνω των 30, να μοιράζονται εξ' ίσου το υπόλοιπο 40%. Με βάση την ηλικία προσχώρησης, μπορούμε να συμπεράνουμε ότι υπάρχει μία τεράστια επίδραση του Πανεπιστημίου στην υπόθεση του Ελεύθερου Λογισμικού. Αυτό δεν αποτελεί έκπληξη, αφού σύμφωνα με όσα είδαμε στο κεφάλαιο περί ιστορίας, πριν καν ακόμη το Ελεύθερο Λογισμικό καταστεί γνωστό με αυτό το όνομα, ήταν ήδη στενά συνδεδεμένο με την Ανώτατη εκπαίδευση. Ακόμη και σήμερα, οι ομάδες φοιτητών-χρηστών και τα Πανεπιστήμια, συνεχίζουν να οδηγούν τη χρήση και την επέκταση του Ελεύθερου Λογισμικού. Άρα, δεν είναι απορίας άξιον ότι πάνω από 70% των developers είναι με Πανεπιστημιακό υπόβαθρο. Αυτά τα στοιχεία καθίστανται ακόμη πιο σημαντικά όταν αναλογισθούμε ότι και το υπόλοιπο 30% δεν είναι ακόμη στο Πανεπιστήμιο, διότι είναι ακόμη σε ηλικία σχολείου.

Ακόμη κι' έτσι, είναι και αυτοί εμπλεγμένοι και δεν χαίρουν καθόλου λιγότερης εκτίμησης σε σχέση με τους developers που δεν είχαν ποτέ πρόσβαση σε Ανώτατη Εκπαίδευση, αλλά είναι απλώς ενθουσιώδεις υποστηρικτές της Πληροφορικής.

Ο προγραμματιστής Ελεύθερου Λογισμικού είναι, συνήθως, ανδρικού φύλλου. Οι αριθμοί που μαζέψαμε από διάφορες μελέτες, για την παρουσία γυναικών μέσα στην Κοινότητα, ποικίλλουν ανάμεσα στο 1% και το 3%, εντασσόμενοι στο περιθώριο λάθους. Συγχρόνως, η πλειοψηφία (60%) δηλώνει ότι έχει ένα σύντροφο, ενώ ο αριθμός developers με παιδιά είναι μόλις 16%. Με δεδομένα αυτά τα όρια ηλικιών των προγραμματιστών Ελεύθερου Λογισμικού, αυτά τα στοιχεία συμπίπτουν με αρκετή ακρίβεια με ένα τυχαίο δείγμα, με την έννοια ότι αυτό μπορεί να θεωρηθεί κανονικά κατανομημένο. Αυτός ο μύθος του μοναχικού developer, σύμφωνα με τον οποίο ο ενθουσιασμός για την Πληροφορική αποτελεί το μόνο πράγμα στη ζωή του, αποδεικνύεται, όπως βλέπουμε, εσφαλμένος, και είναι μάλλον μία εξαίρεση παρά ο κανόνας.

4.3. Τι κάνουν οι developers ;

Επαγγελματικά μιλώντας, οι προγραμματιστές Ελεύθερου Λογισμικού περιγράφουν τον εαυτό τους ως μηχανικοί λογισμικού (33%), φοιτητές (21%), προγραμματιστές (11%), σύμβουλοι/εξωτ. συνεργάτες (10%), Καθηγ. Πανεπιστημίου (7%), κλπ. Στην αντίθετη πλευρά του φάσματος, διαπιστώνουμε ότι, συνήθως, δεν είναι στελέχη των Τμημάτων Πωλήσεων ή marketing (περίπου 1%). Είναι ενδιαφέρον να προσέξουμε πόσοι από αυτούς αυτοχαρακτηρίζονται μηχαν. λογισμικού

παρά “προγραμματιστές”, σχεδόν τριπλάσιοι, έχοντας κατά νου, όπως θα δούμε και στο κεφάλαιο για την Τεχνολογία Λογισμικού, ότι η εφαρμογή των κλασσικών τεχνικών Τεχνολογίας Λογισμικού (ακόμη και των πιο σύγχρονων) δεν περιορίζεται, όντως, μόνο στον κόσμο του Ελεύθερου Λογισμικού.

Η διασύνδεση με το Πανεπιστήμιο, που έχει ήδη αποδειχθεί, ξεπροβάλλει και πάλι μέσα σε αυτό το κεφάλαιο. Σχεδόν ο ένας στους τρεις developers είναι φοιτητής ή Καθηγ. Πανεπιστημίου, που δείχνει την τρομερή συνεργασία ανάμεσα στους ανθρώπους κυρίως από την βιομηχανία παραγωγής λογισμικού (τα υπόλοιπα δύο τρίτα) και τον Πανεπιστημιακό τομέα.

Συγχρόνως, κατέστη εφικτό να εντοπισθεί ένα μεγάλο πεδίο ανάμεικτων ειδικοτήτων: ο ένας στους πέντε developers προέρχεται από έναν χώρο εκτός της Πληροφορικής. Αυτό, σε συνδυασμό με την ύπαρξη, επίσης, ενός αντίστοιχου αριθμού προγραμματιστών μη-Πανεπιστημιακής προέλευσης, αντικατοπτρίζει μία τεράστια ποικιλία και πλούτο ενδιαφερόντων, προελεύσεων, και σίγουρα, σύνθεσης των ομάδων ανάπτυξης. Είναι πολύ δύσκολο να βρει κανείς μία σύγχρονη βιομηχανία, αν υποθέσουμε ότι υπάρχει μία τέτοια, με τόσο μεγάλο βαθμό ανομοιογένειας, όπως εκείνη που είδαμε ότι υπάρχει στο Ελεύθερο Λογισμικό.

Επιπρόσθετα από το περίπου 20% των φοιτητών, το 64% των developers είναι κυρίως έμμισθοι υπάλληλοι, ενώ το ποσοστό των αυτοαπασχολούμενων developers είναι στο 14%.

Τέλος, μόλις το 3% δηλώνουν άνεργοι, διαπίστωση που είναι σημαντική, αφού η μελέτη διεξήχθη μετά το σκάσιμο της φούσκας των εταιρειών dotcom.

Σημείωση

Το γεγονός ότι το επιχειρηματικό μοντέλο ανάπτυξης Ελεύθερου Λογισμικού, σε αντίθεση με τα ισχύοντα για το ιδιοταγές λογισμικό, δεν μπορεί να επιτευχθεί μέσω της πώλησης αδειών χρήσης, πυροδοτούσε ανέκαθεν ζωηρές διαμάχες σχετικά με το πως θα μπορούσαν οι προγραμματιστές να κερδίσουν τα προς το ζην. Στις μελέτες στις οποίες αναφερόμαστε σε αυτό το κεφάλαιο, το 50% των developers δήλωσε ότι αποκόμισε μία άμεση είτε έμμεση ανταμοιβή (compensation) για την εμπλοκή τους στο Ελεύθερο Λογισμικό. Πάντως, πολλοί άλλοι δεν είναι τόσο βέβαιοι. Ο Richard Stallman, ιδρυτής του GNU project, όταν ρωτήθηκε τι θα πρέπει να κάνει ένας developer Ελεύθερου Λογισμικού για να βγάλει λεφτά, απάντησε ότι μπορεί να δουλέψει σαν σερβιτόρος.

4.4. Γεωγραφική κατανομή

Η πρόσβαση σε γεωγραφικά δεδομένα για τους developers, αποτελεί ένα θέμα που θα πρέπει να προσεγγίσουμε με πιο επιστημονικό τρόπο. Το πρόβλημα με την έρευνα που παρουσιάζουμε σε αυτό το κεφάλαιο, είναι ότι καθώς βασίζεται σε μελέτες μέσω Διαδικτύου, που είναι ανοιχτές σε οποιονδήποτε επιθυμεί να συμμετάσχει, η συμμετοχή εξαρτάται σε μεγάλο βαθμό από τα sites προς τα οποία εστάλη, και από τον τρόπο με τον οποίο ανακοινώθηκε. Για να είμαστε ακριβείς, θα πρέπει να τονίσουμε ότι αυτές οι μελέτες δεν στόχευαν να είναι αντιπροσωπευτικές, αλλά μάλλον στο να αποκομίσουν τις απαντήσεις/γνώμες του μεγαλύτερου δυνατού μέρους των developers Ελεύθερου Λογισμικού.

Εν πάσει περιπτώσει, θα μπορούσαμε να αποτολμήσουμε τη διατύπωση μερικών υποθέσεων εργασίας σχετικά με αυτό το θέμα, γνωρίζοντας ότι τα δεδομένα αυτά δεν είναι και τόσο αξιόπιστα όσο τα προηγούμενα, και επομένως, το περιθώριο σφάλματος μπορεί να είναι πολύ μεγαλύτερο. Εκείνο που μοιάζει να είναι σίγουρο είναι ότι οι περισσότεροι developers Ελεύθερου Λογισμικού προέρχονται από βιομηχανοποιημένες χώρες, και ότι η παρουσία developers από τις λεγόμενες χώρες του Τρίτου κόσμου είναι σπάνια. Συνεπώς, δεν πρέπει να εκπλήσσει ότι ο χάρτης των developers του Debian project (<http://www.debian.org/devel/developers.loc>) [187], για παράδειγμα, αντικατοπτρίζει τις φωτογραφίες της Γης κατά τη νύχτα: εκεί όπου υπάρχει φως - βλέπε "εκεί όπου

υπάρχει βιομηχανικός πολιτισμός" – εκεί τείνουν να συγκεντρώνονται και οι developers Ελεύθερου Λογισμικού. Αυτό, που μπορεί να μοιάζει λογικό από πρώτη σκοπιά, είναι σε αντίθεση με τις δυνητικές ευκαιρίες που το Ελεύθερο Λογισμικό μπορεί να προσφέρει στις χώρες του Τρίτου Κόσμου.

Μπορούμε να βρούμε ένα τυπικό παράδειγμα στον ακόλουθο πίνακα, που δείχνει τις πιο συχνές χώρες προέλευσης των developers του Debian project, τα τελευταία 4 χρόνια. Υπάρχει μία ξεκάθαρη τάση προς αποκέντρωση του project, που τεκμαίρεται από το γεγονός ότι ο ρυθμός αύξησης του αριθμού προγραμματιστών από τις ΗΠΑ, τη χώρα που συνεισφέρει περισσότερο, είναι χαμηλότερος από το μέσο όρο. Το γεγονός για το Ελεύθερο Λογισμικό είναι ότι, γενικά, οι διάφορες χώρες κατάφεραν να διπλασιάσουν τον αριθμό των εθελοντών, από το 1999 ως το 2003, και η Γαλλία, που πέτυχε να πεναπλασιάσει την παρουσία της, αποτελεί το πιο ξεκάθαρο παράδειγμα σε αυτό το θέμα. Αν λάβουμε υπ' όψη μας ότι το Debian έκανε τα πρώτα του βήματα στην Αμερικανική ήπειρο (ειδικότερα στις ΗΠΑ και τον Καναδά), τότε μπορούμε να αντιληφθούμε ότι, κατά τα τελευταία λίγα χρόνια, το project εξευρωπαϊσθηκε. Εκτιμάμε ότι το επόμενο βήμα θα είναι η περιζήτητη παγκοσμιοποίηση, με την ενσωμάτωση των Νοτιοαμερικανικών, των Αφρικανικών, και των Ασιατικών χωρών (με την εξαίρεση της Κορέας και της Ιαπωνίας, που ήδη υπεραντιπροσωπεύονται), αν και τα στοιχεία που έχουμε στη διάθεσή μας (δύο συνεργάτες από την Αίγυπτο, την Κίνα ή την Ινδία, και ένας στο Μεξικό, την Τουρκία ή την Κολομβία, κατά τον Ιούνιο του 2003), δεν φαίνονται και τόσο ενθαρρυντικά προς αυτή την κατεύθυνση.

Σε όλο τον κόσμο του Ελεύθερου Λογισμικού (και όχι μόνο στην περίπτωση του Debian), υπάρχει μία εκτεταμένη διαμάχη σχετικά με την υπεροχή της Ευρώπης ή των ΗΠΑ. Όλες οι μελέτες έδειξαν ότι η παρουσία Ευρωπαίων προγραμματιστών είναι ελαφρώς μεγαλύτερη από εκείνη των Βορειοαμερικανών, ένα αποτέλεσμα όμως που μετριάζεται από το γεγονός ότι ο συνολικός Ευρωπαϊκός πληθυσμός είναι μεγαλύτερος από τον Αμερικανικό. Συνεπώς, πρόκειται για έναν πόλεμο αριθμών, αφού ο αριθμός developers ανά κεφαλή είναι μεγαλύτερος για τους Βορειοαμερικανούς. Αν εξετάσουμε και τον αριθμό των ανθρώπων με πρόσβαση στο Internet, αντί για τον συνολικό πληθυσμό, τότε η Ευρώπη υπερτερεί κα πάλι.

Από άποψη χωρών, οι περιοχές με τα υψηλότερα επίπεδα διείσδυσης (implantation), δηλ. αυτό που προκύπτει αν διαιρέσουμε τον αριθμό των developers δια τον συνολικό πληθυσμό) είναι η Βόρεια Ευρώπη (Φινλανδία, Σουηδία, Νορβηγία, Δανία και Ισλανδία) και η Κεντρική Ευρώπη (Χώρες Benelux, Γερμανία και Δημοκρατία της Τσεχίας), ακολουθούμενες από Αυστραλία, Καναδά, Νέα Ζηλανδία και τις ΗΠΑ. Η ζώνη της Μεσογείου, πάντως, παρά τη σημασία της σε απόλυτους αριθμούς (χάρη στους μεγάλους πληθυσμούς της Γαλλίας, της Ιταλίας και της Ισπανίας), βρίσκεται κάτω από τον μέσο όρο.

Πίνακας 1. Οι χώρες με τον μεγαλύτερο αριθμό Debian developers

Χώρα	01/07/ 1999	01/07/ 2000	01/07/ 2001	01/07/ 2002	20/06/2003
ΗΠΑ	162	169	256	278	297
Γερμανία	54	58	101	121	136
Μ. Βρετανία	34	34	55	63	75
Αυστραλία	23	26	41	49	52
Γαλλία	11	11	24	44	51
Καναδάς	20	22	41	47	49
Ισπανία	10	11	25	31	34
Ιαπωνία	15	15	27	33	33
Ιταλία	9	9	22	26	31
Ολλανδία	14	14	27	29	29
Σουηδία	13	13	20	24	27

4.5. Αφοσίωση

Ο αριθμός ωρών που οι developers Ελεύθερου Λογισμικού αφιερώνουν στην ανάπτυξη τέτοιου Ελεύθερου Λογισμικού, παραμένει ένας από τους μεγάλους άγνωστους. Θα πρέπει, επίσης, να τονίσουμε ότι αυτή είναι μία από τις κύριες διαφορές με το εμπορικό λογισμικό που παράγει μία εταιρεία, όπου ο αριθμός των συμμετεχόντων στην ομάδα ανάπτυξης καθώς και ο χρόνος που επενδύθηκε από την κάθε ομάδα για την ανάπτυξη, είναι απολύτως γνωστά στοιχεία. Ο χρόνος που οι developers αφιερώνουν στο Ελεύθερο Λογισμικό μπορεί να εκληφθεί ως ένας έμμεσος δείκτης του επιπέδου του επαγγελματισμού τους. Πριν δείξουμε τα τρεχόντως διαθέσιμα στοιχεία, είναι σημαντικό να σημειώσουμε ότι αυτά αποκτήθηκαν μέσω εκτιμήσεων που έδωσαν οι developers σε διάφορα μελέτες με ερωτηματολόγια, και έτσι, πέραν των ενδογενών ανακρίβειών αυτού του είδους συλλογής δεδομένων, θα πρέπει να συνυπολογίσουμε και ένα περιθώριο σφάλματος που έχει σχέση με το πως ο κάθε developer εννοεί και ορίζει τον “χρόνο ανάπτυξης”. Συνεπώς, είναι βέβαιο ότι πολλοί developers δεν θα συμπεριλάβουν το χρόνο που αφιέρωσαν στην ανάγνωση e-mails (ή, ίσως και να το πράξουν) και να παράσχουν διευκρινίσεις μόνον σχετικά με τον χρόνο που αφιέρωσαν στον προγραμματισμό και την αποσφαλμάτωση (debugging). Άρα λοιπόν, όλοι οι αριθμοί που παραθέτουμε πιο κάτω, πρέπει να ερμηνευθούν με τις απαραίτητες επιφυλάξεις.

Η μέχρι τώρα διεξαχθείσα έρευνα, δείχνει πως ο κάθε developer λογισμικού αφιερώνει 11 ώρες την εβδομάδα, κατά μέσον όρο (*"Motivation of software developers in open source projects: an internet-based survey of contributors to the Linux kernel"*, 2003) [143]. Πάντως, αυτός ο αριθμός μπορεί να είναι παραπλανητικός, αφού υπάρχει μία τεράστια διακύμανση στον χρόνο που επενδύουν οι διάφοροι developers λογισμικού. Στη μελέτη *"Free/ Libre and Open Source Software, Survey and study, part IV: "Survey of developers"*, 2002 [126], το 22.5% των ερωτηθέντων δήλωσαν ότι η συνεισφορά τους ήταν λιγότερο από δύο ώρες την εβδομάδα, και αυτός ο αριθμός αυξήθηκε στο 26.5% για εκείνους που αφιέρωσαν από δύο ως πέντε ώρες την εβδομάδα. Το 21.0% ξόδεψε ανάμεσα στις έξι και τις δέκα ώρες, ενώ το 14.1% ξόδεψε ανάμεσα στις 11 και τις 20 ώρες την εβδομάδα. Το 9.2% υποστήριξε ότι ο χρόνος που ξόδεψαν αναπτύσσοντας Ελεύθερο Λογισμικό ήταν ανάμεσα στις 20 και τις 40 ώρες την εβδομάδα, και το 7.1% επένδυσαν πάνω από 40 ώρες εβδομαδιαίως.

Πίνακας 2. Χρονοώρες αφιέρωσης ανά εβδομάδα

Ωρες ανά εβδομάδα	% Ποσοστό
Λιγότερο από 2 ώρες	22.5%
Ανάμεσα στις 2 και τις 5 ώρες	26.1%
Ανάμεσα στις 5 και 10 ώρες	21.0%
Ανάμεσα στις 10 και 20 ώρες	14.1%
Ανάμεσα στις 20 και 40 ώρες	9.2%
Περισσότερο από 40 ώρες	7.1%

Σημείωση

Εκτός από το να δείχνει το επίπεδο του επαγγελματισμού των ομάδων ανάπτυξης Ελεύθερου Λογισμικού, ο χρόνος που επένδυσαν σε ώρες, συνιστά μία βαρύνουσα παράμετρο όταν πρέπει να γίνουν εκτιμήσεις κόστους και συγκρίσεις με τα ιδιοταγή κλειστά μοντέλα ανάπτυξης που απαντώνται στον επιχειρηματικό και βιομηχανικό κόσμο. Σε ό,τι αφορά το Ελεύθερο Λογισμικό, αυτό που έχουμε, προς το παρόν, είναι μόνο τα τελικά προϊόντα (νέα λογισμικά που βγαίνουν στην κυκλοφορία, συγχρονισμός του νέου κώδικα στις διάφορες εκδόσεις...), πράγμα που δεν μας επιτρέπει να γνωρίζουμε πόσο χρόνο αφιέρωσε ο developer για να το επιτύχει.

Μία ανάλυση αυτών των αριθμών μας πληροφορεί ότι περίπου το 80% των developers κάνουν αυτές τις εργασίες στον ελεύθερό τους χρόνο, ενώ μόνον ο ένας στους πέντε θα εκτιμούσε ότι ξοδεύει ανάλογο χρόνο σε αυτή τη δραστηριότητα ως επαγγελματίας. Αργότερα, στο κεφάλαιο που πραγματεύεται την Τεχνολογία Λογισμικού, θα δούμε πως αυτά τα δεδομένα αντιστοιχίζονται με τις συνεισφορές των developers, αφού και τα δύο φαίνεται να ακολουθούν τον Νόμο του Pareto (βλέπε Κεφάλαιο 7.6).

4.6. Κίνητρα

Υπήρξαν πολλές εικασίες σχετικά με τα κίνητρα που έχουν οι developers για να αναπτύσσουν Ελεύθερο Λογισμικό, ειδικά όταν αυτό γίνεται κατά τον ελεύθερο χρόνο τους (πράγμα που, όπως είδαμε, ισχύει για περίπου το 80% των developers). Όπως και στις προηγούμενες παραγράφους, εμείς διαθέτουμε μόνο τα δεδομένα που προέκυψαν από τις μελέτες. Αυτός είναι ο λόγος για τον οποίο είναι σημαντικό να κατανοήσουμε ότι οι απαντήσεις δόθηκαν από τους ίδιους τους developers, πράγμα που σημαίνει μπορεί να είναι περισσότερο ή λιγότερο συνεπείς με την πραγματικότητα. Τα ποσοστά που εμφανίζονται παρακάτω υπερβαίνουν αθροιστικά το 100% διότι υπήρχε η δυνατότητα, για τους συμμετέχοντες, να επιλέγουν περισσότερες απαντήσεις.

Σε κάθε περίπτωση, από τις απαντήσεις τους φαίνεται ότι οι περισσότεροι θέλουν να μάθουν και να αναπτύξουν νέες ικανότητες (περίπου το 80%) και ότι πολλοί το κάνουν για να μοιραστούν τις γνώσεις και τις ικανότητες (το 50%) ή για να συμμετάσχουν σε ένα νέο σχήμα συνεργασίας (περίπου το ένα τρίτο). Τα πρώτα στοιχεία δεν αποτελούν έκπληξη, καθότι ένας επαγγελματίας που κατέχει παραπάνω γνώση θα είναι περισσότερο περιζήτητος από έναν με λιγότερη. Πάντως, δεν είναι και τόσο εύκολο να ερμηνευθούν τα δεύτερα δεδομένα και θα μπορούσε να πει κανείς ότι μοιάζουν σχεδόν να αντιφάσκουν με την άποψη του Nikolai Bezroukov, στο "A second look at the cathedral and the bazaar" (December, 1998) [91] ότι οι ηγέτες των projects Ελεύθερου Λογισμικού είναι προσεκτικοί να μην μοιράζονται όλες τις πληροφορίες που κατέχουν, για να μπορούν να διαγωνίζονται την δύναμή τους. Στο εντωμεταξύ, η τρίτη πιο συχνή επιλογή αντικατοπτρίζει, αναμφισβήτητα, τον ενθουσιασμό των developers για τον τρόπο με τον οποίο δημιουργείται το Ελεύθερο Λογισμικό γενικότερα. Είναι δύσκολο να βρει κανείς μία βιομηχανία, όπου μία ομάδα χαλαρά οργανωμένων εθελοντών μπορεί -τεχνολογικά μιλώντας- να σταθεί απέναντι στους γίγαντες του τομέως αυτού.

Παρ' ότι η "κλασσική" θεωρία υποστηρίζει ότι ο λόγος που οι developers Ελεύθερου Λογισμικού ξοδεύουν τον χρόνο τους συνεισφέροντας σε αυτό το είδος project είναι η καλή φήμη και τα έμμεσα οικονομικά οφέλη σε μεσομακροπρόθεσμο ορίζοντα, φαίνεται, όμως, ότι οι ίδιοι οι developers διαφωνούν με αυτές τις θέσεις. Μόνο το 5% των ερωτηθέντων απάντησε ότι αναπτύσσουν Ελεύθερο Λογισμικό για να βγάλουν λεφτά, ενώ ο αριθμός εκείνων που το πράττουν για "χτίσουν" μία καλή φήμη ανεβαίνει στο 9%, πολύ μακριά από τις απαντήσεις που δόθηκαν στην προηγούμενη παράγραφο. Σε κάθε περίπτωση, φαίνεται ότι η περαιτέρω διερεύνηση των κινήτρων των developers για να γίνουν μέλη της Κοινότητας του Ελεύθερου Λογισμικού, αποτελεί έναν κεφαλαιώδους σημασίας στόχο, προς τον οποίο οι Κοινωνιολόγοι και οι Ψυχολόγοι θα πρέπει να στραφούν στο εγγύς μέλλον.

4.7. Ηγετική ικανότητα (Leadership)

Η απόκτηση καλής φήμης και η απόδειξη ηγετικής ικανότητας, συνιστούν 2 από τα χαρακτηριστικά που χρησιμοποιούνται για να εξηγηθεί η επιτυχία του Ελεύθερου Λογισμικού, και ειδικά, το μοντέλο του bazaar, όπως θα δούμε στο κεφάλαιο για την Τεχνολογία Λογισμικού. Σύμφωνα με όσα είδαμε σε ένα άλλο κεφάλαιο, για τις άδειες λογισμικού, υπάρχουν μερικές διαφορές ανάμεσα στις Ελεύθερες Άδειες λογισμικού και τις αντίστοιχές τους, στο χώρο των γραπτών κειμένων. Αυτές οι διαφορές προκύπτουν από τον τρόπο διασφάλισης των δικαιωμάτων του συγγραφέως (authorship) και την πιο έντονα βαρύνουσα γνώμη του συγγραφέως σχετικά με το κείμενο, σε σύγκριση με εκείνο που ισχύει για τα προγράμματα λογισμικού.

Στη μελέτη *Free/libre and open source software. Survey and study, part IV: "Survey of developers"* (2002) [126], είχε συμπεριληφθεί μία ερώτηση που ζητούσε από τους developers να δηλώσουν ποια άτομα γνώριζανε, αν και όχι απαραίτητα σε προσωπικό επίπεδο, από έναν κατάλογο.

Τα αποτελέσματα, που φαίνονται στον πίνακα 3, δείχνουν ότι αυτοί οι άνθρωποι μπορούν να καταταχθούν σε τρεις, σαφώς ξεχωριστές, κατηγορίες:

Πίνακας 3. *Βαθμός συνειδητοποίησης για την ύπαρξη σημαντικών developers*

<i>Developer</i>	<i>Γνωστός σε ποσοστό χρηστών</i>
Linus Torvalds	96.5%
Richard Stallman	93.3%
Miguel de Icaza	82.1%
Eric Raymond	81.1%
Bruce Perens	57.7%
Jamie Zawinski	35.8%
Mathias Ettrich	34.2%
Jörg Schilling	21.5%
Marco Pesenti Gritti	5.7%
Bryan Andrews	5.6%
Guenter Bartsch	3.5%
Arpad Gereoffy	3.3%
Martin Hoffstede	2.9%
Angelo Roulini	2.6%
Sal Valliger	1.2%

- Μία πρώτη ομάδα ανθρώπων, με ξεκάθαρα φιλοσοφικά και ιστορικά χαρακτηριστικά, μέσα στον κόσμο του Ελεύθερου Λογισμικού (αν και, όπως γνωρίζουμε, έχουν επίσης και αξιόλογες τεχνικές ικανότητες):

- 1) Ο Linus Torvalds. Ο δημιουργός του πυρήνα Linux, το πλέον χρησιμοποιούμενο Ελεύθερο Λειτουργικό Σύστημα, και συνδημιουργός του *Just for fun: the story of an accidental revolutionary* [217].
- 2) Ο Richard Stallman. Ιδεολόγος και ιδρυτής του *Free Software Foundation* (Ιδρυμα Ελεύθερου Λογισμικού) και developer σε διάφορα GNU projects. Συγγραφέας διαφόρων σημαντικών δοκιμίων (essays) σχετικά με το Ελεύθερο Λογισμικό ("*Why free software is better than open source*", 1998 [206], "*Copyleft: pragmatic idealism*", 1998 [205], "*The GNU Project*" [208] and "*The GNU Manifesto*", 1985 [117]).
- 3) Ο Miguel de Icaza. Συνιδρυτής του *GNOME project* και της *Ximian Inc.*, και developer στο *GNOME* και το *MONO*.
- 4) Ο Eric Raymond. Ο προωθητής του *Open Source Initiative*, συγγραφέας του "*The cathedral and the bazaar*" [192] και βασικός δημιουργός του *fetchmail*.
- 5) Ο Bruce Perens. Πρώην ηγέτης του *Debian project*, προωθητής (προσηλυτισθείς) του *Open Source Initiative* και developer του εργαλείου *e-fence*.
- 6) Ο Jamie Zawinsky. Τέως developer του *Mozilla* και διάσημος για μία επιστολή του το έτος 1999, με την οποία εγκατέλειπε το *Mozilla project*, υποστηρίζοντας ότι το μοντέλο που ακολουθούσαν δεν θα απέφερε ποτέ καρπούς ("*Resignation and postmortem*", 1999) [237].
- 7) Ο Mathias Ettrich. Ιδρυτής του *KDE* και developer του *LyX* και άλλων.

- Μία δεύτερη ομάδα που αποτελείται από developers. Αυτή η μελέτη έλαβε υπ' όψη τα ονόματα των σημαντικότερων developers, των 6 πλέον δημοφιλών projects, σύμφωνα με την ιστοσελίδα *FreshMeat*, απ' όπου μπορεί κανείς να κατεβάσει Ελεύθερο Λογισμικό. Μπορούμε να δούμε ότι (με την εξαίρεση του Linus Torvalds, για προφανείς λόγους, και του Jörg Schilling), το επίπεδο επήγνωσης αυτών των developers είναι μικρό:

- 1) Jörg Schilling, δημιουργός, ανάμεσα σε άλλες εφαρμογές, του *CDrecord*.
- 2) Marco Pesenti Gritti, ο κύριος developer του *Galeon*.
- 3) Bryan Andrews, developer του *Apache Toolbox*.
- 4) Guenther Bartsch, δημιουργός του *Xine*.
- 5) Arpad Gereoffy, developer του *MPEGPlayer*.

- Μία τρίτη ομάδα, που αποτελείται από από τα ονόματα των τριών τελευταίων “ανθρώπων” στον κατάλογο. Αυτά τα ονόματα επινοήθηκαν από την ομάδα έρευνας, για να ελέγξουν το περιθώριο σφάλματος.

Μπορούμε να αντλήσουμε δύο συμπεράσματα από αυτά τα αποτελέσματα: το πρώτο είναι ότι μπορούμε να θεωρήσουμε το περιθώριο σφάλματος ως μικρό (λιγότερο από 3%), και το δεύτερο

είναι ότι οι περισσότεροι developers των δημοφιλέστερων εφαρμογών του Ελεύθερου Λογισμικού, είναι τόσο διάσημοι όσο και άνθρωποι που δεν υπάρχουν. Αυτά τα στοιχεία θα πρέπει να οδηγήσουν όσους υποστηρίζουν πως ένας από τους κύριους λόγους ανάπτυξης Ελεύθερου Λογισμικού είναι η αναζήτηση φήμης, να το ξανασκεφθούν διπλά.

4.8. Περίληψη και συμπεράσματα

Αυτό το κεφάλαιο προσπάθησε να ρίξει λίγο φως στο ευρέως άγνωστο θέμα των ανθρώπων που αφιερώνουν το χρόνο τους στο Ελεύθερο Λογισμικό. Σε γενικές γραμμές, μπορούμε να πούμε πως ένας developer Ελεύθερου Λογισμικού είναι ένας νέος άνδρας, με Πανεπιστημιακό πτυχίο (είτε καθ' οδόν προς την απόκτησή του). Η σχέση ανάμεσα στον κόσμο του Ελεύθερου Λογισμικού και τα Πανεπιστήμια (φοιτητές και καθηγητές) είναι πολύ στενή, αν και, ακόμη, υπερισχύει ο developer που δεν έχει καμία σχέση με τον Ακαδημαϊκό χώρο.

Από την σκοπιά των ωρών αφιέρωσης, δείξαμε ότι υπάρχει μία τεράστια ανομοιογένεια, παρόμοια με όσα προβλέπει ο Νόμος του Pareto. Τα κίνητρα των developers, σύμφωνα με την δική τους άποψη, απέχουν πολύ από το να είναι χρηματικά και εγωκεντρικά, όπως οι Οικονομολόγοι και οι Ψυχολόγοι εισηγούνται, και έχουν να κάνουν πολύ περισσότερο με το να μοιράζονται και με το να μαθαίνουν. Τέλος, δείξαμε ένα πίνακα με τις πιο σημαντικές προσωπικότητες του κόσμου του Ελεύθερου Λογισμικού (περιλαμβάνοντας και άλλους, όχι και τόσο γνωστούς, όπως είδαμε) και αποδείξαμε ότι η κατάκτηση φήμης, μέσα στην τεράστια Κοινότητα του Ελεύθερου Λογισμικού, τείνει να συναρτάται με πολλά περισσότερα από την απλή συγγραφή κώδικα για μία επιτυχημένη εφαρμογή Ελεύθερου Λογισμικού.

5. Οικονομία

"Res publica non dominetur." (Ελεύθερη μετάφραση: "Τα Δημόσια πράγματα δεν έχουν ιδιοκτήτη.")
Εμφανίστηκε σε μία διαφήμιση της IBM για το Linux (2003)

Αυτό το κεφάλαιο εξετάζει κάποιες οικονομικές πλευρές που σχετίζονται με το Ελεύθερο Λογισμικό. Θα αρχίσουμε δείχνοντας πως χρηματοδοτούνται τα projects Ελεύθερου Λογισμικού (όταν τουλάχιστον χρηματοδοτούνται, αφού σε πολλές περιπτώσεις βασίζονται μόνον σε προσπάθειες και πόρους που συνεισφέρονται εθελοντικά). Στο επόμενο, θα εξετάσουμε τα βασικά επιχειρηματικά μοντέλα που εφαρμόζουν οι εταιρείες που σχετίζονται άμεσα με το Ελεύθερο Λογισμικό. Το κεφάλαιο τελειώνει με μία μικρή μελέτη για τη σχέση ανάμεσα στο Ελεύθερο Λογισμικό και τα μονοπώλια στο χώρο της βιομηχανίας λογισμικού.

5.1. Χρηματοδότηση των projects Ελεύθερου Λογισμικού

Το Ελεύθερο Λογισμικό αναπτύσσεται με πολλούς διαφορετικούς τρόπους και χρησιμοποιεί μηχανισμούς προσπορισμού κεφαλαίων που παρουσιάζουν τεράστια ποικιλία, από περίπτωση σε περίπτωση. Κάθε Ελεύθερο project διαθέτει το δικό του τρόπο αυτοχρηματοδότησης, ξεκινώντας από εκείνον που βασίζεται αποκλειστικά στην παρουσία εθελοντών developers και χρησιμοποιεί μόνον χρηματικά ποσά που δωρήθηκαν αλτρουιστικά, και φθάνοντας μέχρις εκείνον που διεξάγεται από μία εταιρεία που τιμολογεί τα κόστη της κατά 100%, προς μία οργάνωση, η οποία ενδιαφέρεται για την ανάπτυξη του εν λόγω project.

Σε αυτό το κεφάλαιο, θα εστιάσουμε την προσοχή μας σε εκείνα τα projects για τα οποία υφίσταται εξωτερική χρηματοδότηση και όπου η δουλειά δεν είναι καθ' ολοκληρίαν εθελοντική. Σε αυτές τις

περιπτώσεις, υπάρχει συχνά κάποια μορφή ροής εισερχόμενου χρήματος, από μία εξωτερική πηγή που είναι υπεύθυνη για την παροχή κεφαλαίων, με σκοπό την ανάπτυξή του. Έτσι, το Ελεύθερο Λογισμικό που παράγεται με αυτόν τον τρόπο, μπορεί να θεωρηθεί, εντός κάποιων ορίων, ως το προϊόν αυτής της εξωτερικής χρηματοδότησης. Αυτός είναι και ο λόγος για τον οποίο είναι συχνό να μπορεί αυτή η εξωτερική πηγή να αποφασίζει (τουλάχιστον εν μέρει) πως θα ξοδευτούν τα χρήματα και σε ποια πράγματα.

Υπό μίαν έννοια, λοιπόν, αυτής της μορφής η εξωτερική χρηματοδότηση για τα Ελεύθερα projects, μπορεί να θεωρηθεί σαν μίας μορφής Χορηγία, αν και αυτή η Χορηγία δεν έχει κανένα λόγο να είναι ανιδιοτελής (και συνήθως δεν είναι). Στα επόμενα κεφάλαια, θα συζητήσουμε τις πιο συνηθισμένες μορφές εξωτερικής χρηματοδότησης. Πάντως, ενώ θα μαθαίνουμε γι' αυτές, θα πρέπει να θυμόμαστε ότι αυτοί είναι μόνον κάποιοι από τους τρόπους με τους οποίους τα projects Ελεύθερου Λογισμικού μπορούν να αντλήσουν κεφάλαια. Αλλά υπάρχουν και άλλοι, μεταξύ των οποίων ο σημαντικότερος είναι (όπως είδαμε στο Κεφάλαιο 4) η προσφορά εργασίας εκ μέρους πολλών εθελοντών developers.

5.1.1. Δημόσια χρηματοδότηση

Μία πολύ ειδική μορφή χρηματοδότησης για τα Ελεύθερα projects είναι η δημόσια χρηματοδότηση. Η χρηματοδοτούσα Αρχή μπορεί να είναι απ' ευθείας μία Κυβέρνηση (τοπικοί Δήμοι ή Περιφέρειες, εθνικές κυβερνήσεις ή ακόμη και υπερεθνικοί οργανισμοί) είτε ένα Δημόσιο θεσμικό Νομικό πρόσωπο (π.χ. ένα Ίδρυμα). Σε αυτές τις περιπτώσεις, η χρηματοδότηση τείνει να μοιάζει με εκείνη που προβλέπεται για projects έρευνας και ανάπτυξης, και είναι πράγματι συχνό να προέρχεται από Δημόσιους Οργανισμούς που προωθούν την έρευνα και την ανάπτυξη. Κανονικά, ο Οργανισμός που χρηματοδοτεί δεν θα προσπαθήσει να ανακτήσει την επένδυση (ή, τουλάχιστον, όχι ευθέως), αν και συνήθως έχει ξεκάθαρους στόχους (όπως η προώθηση και δημιουργία ενός βιομηχανικού- παραγωγικού ιστού, είτε η προώθηση μίας συγκεκριμένης τεχνολογίας, ή κάποιου τύπου εφαρμογής, κλπ.).

Στις περισσότερες από αυτές τις περιπτώσεις, δεν υπάρχει εμφανής χρηματοδότηση για προϊόντα ή υπηρεσίες που να συνδέονται με το Ελεύθερο Λογισμικό, παρά μάλλον αυτό τείνει να είναι το υποπροϊόν ενός συμβολαίου με άλλους γενικότερους στόχους. Για παράδειγμα, στα πλαίσια των προγραμμάτων Έρευνας που διαθέτει, η Ευρωπαϊκή Επιτροπή χρηματοδοτεί διάφορα projects που έχουν ως στόχο την βελτίωση της Ευρωπαϊκής ανταγωνιστικότητας σε ορισμένους τομείς. Μερικά από αυτά τα projects, προβλέπουν ως τμήμα του σκοπού τους, τη χρήση, τη βελτίωση και τη δημιουργία Ελεύθερου Λογισμικού, εντός των πλαισίων της έρευνας (σαν ένα εργαλείο έρευνας ή σαν ένα προϊόν που προέρχεται από αυτήν).

Τα κίνητρα γι' αυτή τη μορφή χρηματοδότησης είναι ποικίλα, αλλά μπορούμε να διακρίνουμε τα εξής:

1) Επιστημονικά. Πρόκειται για το πιο σύνηθες κίνητρο στην περίπτωση των projects με δημόσια χρηματοδότηση. Αν και ο στόχος δεν είναι η παραγωγή λογισμικού αλλά μάλλον η διερεύνηση ενός συγκεκριμένου πεδίου (είτε έχει σχέση με την Πληροφορική/ IT, είτε όχι), είναι πιθανό να απαιτεί την ανάπτυξη κάποιων προγραμμάτων, σαν εργαλεία για την επίτευξη των σκοπών του project. Συνήθως το project δεν ενδιαφέρεται για την εμπορικοποίηση αυτών των εργαλείων, ή μπορεί να ενδιαφέρεται ενεργώς για να μπορέσουν και άλλες ομάδες ερευνητών να τα χρησιμοποιήσουν και να τα βελτιώσουν. Σε αυτές τις περιπτώσεις, είναι πολύ συχνό να τα διανέμει υπό μορφή Ελεύθερου Λογισμικού. Κατ' αυτόν τον τρόπο, η ομάδα που διεξήγαγε την έρευνα αφιέρωσε, εν μέρει, κάποια κεφάλαια για την παραγωγή αυτού του Λογισμικού, και έτσι μπορούμε να πούμε ότι

αυτό αναπτύχθηκε με δημόσια χρηματοδότηση.

2) Προώθηση προδιαγραφών. Η χρήση μίας υλοποίησης αναφοράς (*reference implementation*) είναι ένας από τους καλύτερους τρόπους για την προώθηση των προδιαγραφών. Σε πολλές περιπτώσεις, αυτό συνεπάγεται την ύπαρξη προγραμμάτων που είναι μέρος της προαναφερθείσας υλοποίησης (ή, αν οι προδιαγραφές αυτές αναφέρονται στο χώρο του Λογισμικού, τότε από μόνα τους θα αποτελούν την υλοποίηση). Για να είναι χρήσιμη η υλοποίηση αναφοράς στην προώθηση κάποιων προδιαγραφών, πρέπει να είναι εύκολα διαθέσιμη, τουλάχιστον με τέτοιο τρόπο ώστε να μπορεί να ελέγχεται η διαλειτουργικότητα, εκ μέρους όλων εκείνων που επιθυμούν να αναπτύξουν προϊόντα που να συμμορφώνονται με αυτές τις προδιαγραφές. Σε πολλές περιπτώσεις, είναι επίσης προτιμώτερο για τους κατασκευαστές να μπορούν να προσαρμόζουν απ' ευθείας την υλοποίηση αναφοράς (*reference implementation*), για να μπορούν να τη χρησιμοποιούν στα προϊόντα τους, εφ' όσον το επιθυμούν.

Αυτός, για παράδειγμα, ήταν ο τρόπος με τον οποίο αναπτύχθηκαν τα πρωτόκολλα για το Διαδίκτυο, που τώρα έχουν καταστεί ο γενικός κανόνας. Σε αυτές τις περιπτώσεις, το να κυκλοφορήσουν οι υλοποιήσεις αναφοράς υπό μορφή Ελεύθερου Λογισμικού, μπορεί να συνεισφέρει σε τεράστιο βαθμό στην προώθησή τους. Για μια ακόμη φορά, το Ελεύθερο Λογισμικό είναι εδώ ένα υποπροϊόν, στην περίπτωση της προώθησης των προδιαγραφών. Κανονικά, ο φορέας που είναι υπεύθυνος γι' αυτή την προώθηση είναι ένας Δημόσιος Οργανισμός (αν και ορισμένες φορές, μπορεί να πρόκειται και για σύμπραξη/consortium ιδιωτικών εταιρειών).

3) Κοινωνικά. Το Ελεύθερο Λογισμικό αποτελεί ένα πολύ ενδιαφέρον εργαλείο για τη δημιουργία μίας βασικής υποδομής της Κοινωνίας της Πληροφορίας. Οι οργανισμοί που ενδιαφέρονται να χρησιμοποιήσουν Ελεύθερο Λογισμικό για να προωθήσουν τη γενικευμένη (universal) πρόσβαση στην Κοινωνία της Πληροφορίας, μπορούν να χρηματοδοτήσουν τα σχετικά projects (θα πρόκειται, συνήθως, για projects ανάπτυξης νέων εφαρμογών, ή την προσαρμογή των ήδη υπαρχόντων).

Σημείωση

Ένα παράδειγμα δημόσιας χρηματοδότησης για έναν κυρίως κοινωνικό στόχο, είναι η περίπτωση του *gnuLinEx*, που προώθησε η Τοπική Περιφερειακή Αυτοδιοίκηση της *Extremadura* (Ισπανία), για να διευκολύνει την πρόσβαση στην Κοινωνία της Πληροφορίας, κυρίως με την έννοια της μείωσης του ψηφιακού χάσματος. Η Τοπική διοίκηση χρηματοδότησε την ανάπτυξη μίας Διανομής που βασιζόταν στο *Debian*, για να μπορέσει να επιτύχει αυτό το στόχο. Μία άλλη όμοια περίπτωση είναι εκείνη της Γερμανικής κυβέρνησης, που χρηματοδότησε την ανάπτυξη του *GnuPG*, με σκοπό να το καταστήσει εύκολο στη χρήση από τους απλούς χρήστες, έχοντας ως κεντρική ιδέα την προώθηση της χρήσης ασφαλούς ηλεκτρονικού ταχυδρομείου από τους πολίτες.

Η ανάπτυξη του GNAT

Μία διάσημη περίπτωση δημόσιας χρηματοδότησης για την ανάπτυξη ενός Ελεύθερου Λογισμικού είναι η περίπτωση του προγράμματος μεταγλώττισης GNAT. Ο GNAT, ένας μεταγλωττιστής Ada, χρηματοδοτήθηκε από το project Ada 9X του Αμερικανικού Υπουργείου Αμύνης, με σκοπό να φτιαχτεί ένα πρόγραμμα μεταγλώττισης για τη νέα έκδοση της γλώσσας προγραμματισμού Ada (που, αργότερα, μετονομάστηκε σε Ada 95), και την οποία προσπαθούσε να προωθήσει τον καιρό εκείνο. Μία από τις αιτίες που εντοπίστηκαν σχετικά με τις εταιρείες λογισμικού που υιοθέτησαν την πρώτη έκδοση της Ada (την Ada 83), ήταν η καθυστερημένη διάθεση του προγράμματος μεταγλώττισης και το υψηλό του κόστος, όταν επιτέλους δόθηκε στην κυκλοφορία. Προσπάθησαν, λοιπόν, να αποφύγουν να συμβεί το ίδιο με το Ada 95, διασφαλίζοντας τη διάθεση του μεταγλωττιστή σχεδόν ταυτόχρονα με την απόδοση στην κυκλοφορία των νέων προδιαγραφών για τη γλώσσα.

Για να το πετύχουν, η Ada 9X ανέθεσε το project σε μία ομάδα από το Πανεπιστήμιο της Νέας

Υόρκης (NYU), έναντι ενός τιμήματος περίπου 1 εκατομ. δολαρίων, για να δημιουργήσουν μία πρότυπη υλοποίηση του μεταγλωττιστή Ada 95. Χρησιμοποιώντας αυτά τα κεφάλαια, και εκμεταλλευόμενοι την ύπαρξη του GCC (μεταγλωττιστής C του GNU, του οποίου το μεγαλύτερο μέρος του backend χρησιμοποιήθηκε), η ομάδα του NYU δημιούργησε με αποτελεσματικότητα τον πρώτο μεταγλωττιστή Ada 95, που κυκλοφορεί υπό την Άδεια GNU GPL. Ο μεταγλωττιστής σημείωσε τέτοια επιτυχία που, όταν το project περατώθηκε, μερικοί από τους δημιουργούς του ίδρυσαν μία εταιρεία (την Ada Core Technologies), που έκτοτε αναδείχθηκε σε ηγέτη της αγοράς των μεταγλωττιστών και των εργαλείων βοήθειας για την ανάπτυξη προγραμμάτων σε Ada.

Σε αυτό το project είναι σημαντικό να σημειώσουμε τον συνδυασμό στοιχείων έρευνας (πράγματι, αυτό το project συνεισέφερε στην πρόοδο της γνώσης για την δημιουργία front ends και συστημάτων run time, για μεταγλωττιστές γλώσσας τύπου Ada) και προώθησης των προδιαγραφών (πράγμα που ήταν ο πιο ξεκάθαρος στόχος του χρηματοδοτικού οργανισμού).

5.1.2. Ιδιωτική Μη-κερδοσκοπική χρηματοδότηση

Αυτού του είδους η χρηματοδότηση διαθέτει πολλά παρόμοια χαρακτηριστικά με το προηγούμενο είδος, που συνήθως διεξάγεται από Ιδρύματα ή από Μη-Κυβερνητικές οργανώσεις. Το άμεσο κίνητρο σε αυτές τις περιπτώσεις είναι να παραχθεί Ελεύθερο Λογισμικό, προς χρήση σε τομείς που ο χρηματοδοτικός οργανισμός θεωρεί ιδιαίτερα κρίσιμους, αλλά μπορούμε επίσης να διαπιστώσουμε και τα έμμεσα κίνητρα της συνεισφοράς στην επίλυση προβλημάτων (για παράδειγμα, ένα ίδρυμα που προωθεί την έρευνα για κάποιο νόσημα, μπορεί να χρηματοδοτήσει την δημιουργία ενός προγράμματος στατιστικής, για να βοηθήσει στην ανάλυση των πειραματικών ομάδων που χρησιμοποιούνται στο πλαίσιο της έρευνας αυτού του νοσήματος). Σε γενικές γραμμές, τόσο τα κίνητρα, όσο και οι μηχανισμοί γι' αυτό τον τύπο χρηματοδότησης, είναι πολύ παρόμοιοι με εκείνους της δημόσιας χρηματοδότησης, αν και βρίσκονται φυσικά, πάντα, εντός των πλαισίων και των σκοπών του χρηματοδοτικού οργανισμού.

Σημείωση

Πιθανώς, το αρχέτυπο παράδειγμα ενός Ιδρύματος που να προωθεί την ανάπτυξη του Ελεύθερου Λογισμικού, είναι το Ίδρυμα Ελεύθερου Λογισμικού (Free Software Foundation, FSF). Από τα μισά της δεκαετίας του 1980, αυτό το Ίδρυμα αφιερώθηκε στην προώθηση του έργου GNU και, γενικότερα, στην ενθάρρυνση της δημιουργίας Ελεύθερου Λογισμικού. Ακόμη μία ενδιαφέρουσα περίπτωση, αν και σε ένα μάλλον ιδιαίτερο πεδίο, είναι το Ίδρυμα Ανοικτής Βιο-Πληροφορικής (Open Bioinformatics Foundation). Οι σκοποί αυτού του Ιδρύματος περιλαμβάνουν την προώθηση της ανάπτυξης βασικών προγραμμάτων υπολογιστή για την έρευνα σε οποιοδήποτε κλάδο της Βιοπληροφορικής. Σε γενικές γραμμές, καταφέρνει την προώθηση αυτή χρηματοδοτώντας και συνεισφέροντας στη δημιουργία Ελεύθερων προγραμμάτων.

5.1.3. Χρηματοδότηση από όποιον αναζητά βελτιώσεις

Μία άλλη μορφή χρηματοδότησης για την ανάπτυξη Ελεύθερου Λογισμικού, που δεν είναι και τόσο αλτρουιστική, λαμβάνει χώρα όταν κάποιος χρειάζεται να επιφέρει βελτιώσεις σε ένα Ελεύθερο προϊόν. Για παράδειγμα, μπορεί μία εταιρεία να χρειάζεται, για εσωτερική χρήση, ένα κάποιο πρόγραμμα να διαθέτει μία συγκεκριμένη λειτουργικότητα ή να διορθώσει μερικά bugs. Σε αυτές τις περιπτώσεις, είναι συνηθισμένο για την εν λόγω εταιρεία, να αναθέτει την επιθυμητή ανάπτυξη. Πρόκειται συχνά για Ελεύθερο Λογισμικό (είτε διότι η Άδεια του υπό τροποποίηση προγράμματος το επιβάλλει, είτε διότι έτσι επιθυμεί η Εταιρεία).

Η περίπτωση του Corel και του Wine

Κατά το τέλος της δεκαετίας του 1990, η Corel αποφάσισε να προσαρμόσει (*port*) τα προϊόντα της στο GNU/Linux. Κατά τη διάρκεια αυτής της διαδικασίας ανακάλυψε ότι ένα Ελεύθερο πρόγραμμα που να είναι σχεδιασμένο για να διευκολύνει την εκτέλεση των εκτελέσιμων για Windows, μέσα σε περιβάλλον GNU/Linux, θα μπορούσε να συνεισφέρει στην επίτευξη σημαντικής εξοικονόμησης κατά την ανάπτυξη. Αλλά για να το πετύχει, έπρεπε να βελτιωθεί, προσθέτοντας ουσιαστικά την προσομοίωση κάποιας λειτουργικότητας για Windows, που χρησιμοποιούσαν τα προγράμματα της Corel. Έτσι η Corel το ανέθεσε στην Macadamian, που συνεισέφερε τις βελτιώσεις της στο Wine project. Κατ' αυτό τον τρόπο, επωφελήθηκαν τόσο η Corel όσο και το Wine.

5.1.4. Χρηματοδότηση με τα σχετικά οφέλη

Σε αυτή τη μορφή χρηματοδότησης, ο χρηματοδοτικός οργανισμός αποσκοπεί να αποκομίσει οφέλη από τα προϊόντα που σχετίζονται με το πρόγραμμα του οποίου την ανάπτυξη χρηματοδοτεί. Σε αυτές τις περιπτώσεις, κανονικά, τα οφέλη που προσπορίζεται ο χρηματοδοτικός οργανισμός δεν είναι αποκλειστικά, αφού και άλλοι μπορούν να μπουν στην αγορά, και να πουλήσουν αυτά τα σχετικά προϊόντα, αλλά είτε το μερίδιο αγοράς που κατακτά είναι επαρκές για να μην ανησυχεί και τόσο πολύ για τη μοιρασιά της πίτας με τους άλλους, είτε διαθέτει ένα ξεκάθαρο ανταγωνιστικό πλεονέκτημα.

Μερικά παραδείγματα προϊόντων που σχετίζονται με συγκεκριμένο λογισμικό, έχουν ως εξής:

- **Βιβλία.** Η εν λόγω εταιρεία πουλά εγχειρίδια, οδηγούς χρήσης, υλικά για σειρές μαθημάτων, κλπ. που σχετίζονται με το Ελεύθερο πρόγραμμα, του οποίου βοηθά την χρηματοδότηση. Εννοείται ότι και άλλες εταιρείες μπορούν, επίσης, να πουλάνε σχετικά βιβλία, αλλά, κανονικά, η χρηματοδότηση του project θα χαρίσει στην εταιρεία πρώτη πρόσβαση σε κρίσιμους developers, πριν από τον ανταγωνισμό, ή απλώς θα χτίσει μία καλή εικόνα προς την κοινότητα χρηστών του εν λόγω προγράμματος.
- **Hardware.** Αν μία εταιρεία χρηματοδοτεί την ανάπτυξη Ελεύθερων συστημάτων για έναν συγκεκριμένο τύπο hardware, τότε μπορεί εύκολα να αφιερωθεί στην πώληση εκείνου του είδους hardware. Για μία ακόμη φορά, αφού το λογισμικό που αναπτύχθηκε είναι Ελεύθερο, μπορεί να εμφανισθούν ανταγωνιστές που πωλούν τον ίδιο τύπο συσκευών, και που επωφελούνται από την προηγούμενη ανάπτυξη, χωρίς να έχουν συμμετάσχει στην χρηματοδότησή της. Αλλά ακόμη κι' έτσι, η εν λόγω εταιρεία, διαθέτει διάφορα πλεονεκτήματα έναντι των ανταγωνιστών της. Ένα από αυτά μπορεί να είναι η θέση της ως πηγή χρηματοδότησης για το project, που της επιτρέπει να ασκεί επιρροή, με τρόπο τέτοιο που να δίδεται περισσότερη προτεραιότητα σε εκείνου του είδους την ανάπτυξη που την ενδιαφέρει περισσότερο.
- **CD με προγράμματα.** Το καλύτερα γνωστό μοντέλο αυτού του είδους, είναι, πιθανώς, εκείνο των εταιρειών που χρηματοδοτούν την ανάπτυξη κάποιων προγραμμάτων, που μπορούν, στη συνέχεια, να εφαρμόσουν στη δική τους διανομή λογισμικού. Για παράδειγμα, το να διαθέτεις ένα καλό περιβάλλον για την επιφάνεια εργασίας, μπορεί να βοηθήσει πολύ στο να πουλήσεις ένα CD με κάποια διανομή του GNU/Linux και, επομένως, η χρηματοδότηση της ανάπτυξής του θα μπορούσε να αποτελέσει μία καλή επιχειρηματική πρακτική για εκείνον που πουλά τα CDs. Πρέπει να έχουμε κατά νου ότι, υπό αυτή την επικεφαλίδα, η χρηματοδότηση πρέπει να γίνεται με σκοπό την πραγματοποίηση κέρδους, και άρα, ο χρηματοδοτικός οργανισμός πρέπει να αποκομίσει ένα δυνητικό όφελος από αυτήν. Στην πραγματικότητα, πάντως, είναι συνηθισμένο να συνυπάρχει ένας συνδυασμός του κινήτρου του κέρδους και του αλτρουισμού, όταν μία εταιρεία παρέχει κεφάλαια για δημιουργηθεί ένα Ελεύθερο πρόγραμμα, από το οποίο προσδοκά να επωφεληθεί άμεσα.

Σημείωση

Μία πολύ γνωστή περίπτωση συνεισφοράς κεφαλαίων σε ένα project, αν και αρκετά έμμεση, είναι η βοήθεια που δίνει ο εκδοτικός οργανισμός O'Reilly για την ανάπτυξη της Perl. Φυσικά, δεν αποτελεί σύμπτωση ότι η O'Reilly είναι και ένας από τους βασικούς εκδότες θεμάτων που σχετίζονται με την Perl. Σε κάθε περίπτωση, είναι προφανές ότι η O'Reilly δεν έχει αποκλειστικότητα επί της δημοσίευσης βιβλίων αυτού του είδους, και ότι άλλοι εκδοτικοί οίκοι ανταγωνίζονται σε αυτό τον τομέα της αγοράς, με ποικίλους βαθμούς επιτυχίας.

Η VA Software (αρχικά VA Research και έπειτα VA Linux) συνεργάστηκε ενεργά στην ανάπτυξη του πυρήνα Linux. Μέσα από αυτή τη διαδικασία, πέτυχε, μεταξύ άλλων, μία διασφάλιση της συνέχειας, που ήταν ιδιαίτερα κρίσιμης σημασίας για την σχέση της εταιρείας με τους πελάτες της, αφού η κυρίως δραστηριότητά της ήταν η πώληση εξοπλισμού με προεγκατεστημένο το GNU/Linux.

Η Red Hat χρηματοδότησε την ανάπτυξη πολλών τμημάτων του GNOME, ουσιαστικά αποκτώντας ένα περιβάλλον για την επιφάνεια εργασίας της διανομής της, πράγμα που βοήθησε στην αύξηση των πωλήσεών της. Όπως και σε προηγούμενες περιπτώσεις, και άλλοι παραγωγοί διανομών επωφελήθηκαν από αυτή την ανάπτυξη, αν και πολλοί από αυτούς δεν συνεργάστηκαν για το έργο GNOME στην ίδια έκταση σαν τη Red Hat (και λίγοι από αυτούς δεν συνεργάστηκαν καθόλου). Παρά το γεγονός αυτό, η Red Hat αντλεί όφελος από την συνεισφορά της στο GNOME.

5.1.5. Χρηματοδότηση υπό μορφή εσωτερικής επένδυσης

Υπάρχουν εταιρείες που αναπτύσσουν απ' ευθείας Ελεύθερο Λογισμικό, ως μέρος του επιχειρηματικού τους μοντέλου. Για παράδειγμα, μία εταιρεία μπορεί να αποφασίσει να ξεκινήσει ένα νέο Ελεύθερο project, σε ένα πεδίο όπου πιστεύει πως υπάρχουν επιχειρηματικές ευκαιρίες, με το σκεπτικό ότι η αρχική επένδυση, θα αποδώσει σε ένα μετέπειτα στάδιο. Αυτό το μοντέλο θα μπορούσε να θεωρηθεί ως μία ποικιλία του προηγούμενου (της έμμεσης χρηματοδότησης), και τα “σχετικά οφέλη” θα είναι τα πλεονεκτήματα που η εταιρεία αποκομίζει από την ανάπτυξη ενός Ελεύθερου Προγράμματος. Αλλά, αφού σε αυτή την περίπτωση, είναι το ίδιο το ελεύθερο προϊόν που περιμένει κανείς ότι θα φέρει τα οφέλη, μοιάζει σωστό να του αποδοθεί προτεραιότητα.

Αυτή η μορφή χρηματοδότησης δημιουργεί διάφορα επιχειρηματικά μοντέλα. Όταν τα αναλύσουμε (στην παράγραφο 5.2), θα εξηγήσουμε και τα πλεονεκτήματα που συνήθως κερδίζει μία εταιρεία, από αυτό το είδος επένδυσης σε ένα project και ποιες μέθοδοι χρησιμοποιούνται για να καταστεί κερδοφόρο. Αλλά σε κάθε περίπτωση, θα πρέπει να τονίσουμε ότι, μερικές φορές, το εν λόγω λογισμικό μπορεί να αναπτυχθεί απλώς για να ικανοποιήσει τις ανάγκες της ίδιας της εταιρείας, ενώ μπορεί μόνον αργότερα να αποφασίσει η εταιρεία να το κυκλοφορήσει, και ίσως, να ανοίξει και μία νέα επιχειρηματική γραμμή που να βασίζεται σε αυτό.

Σημείωση

Η Digital Creations (τώρα Zope Corporation) είναι μία από τις πιο γνωστές περιπτώσεις εταιρείας αφιερωμένης στην ανάπτυξη Ελεύθερου Λογισμικού, με την προσδοκία να πετύχει μία απόδοση της αρχικής επένδυσης. Το Ελεύθερο project στο οποίο κυρίως επενδύει η Zope, είναι ένας διακομιστής εφαρμογών, ο οποίος σημειώνει κάποια επιτυχία. Η ιστορία της με το Ελεύθερο Λογισμικό άρχισε όταν η τότε Digital Creations αναζητούσε ένα επενδυτικό κεφάλαιο (venture capital) για να αναπτύξει τον διακομιστή των ιδιοταγών (proprietary) εφαρμογών της, περί το 1998. Μία από τις ομάδες που ενδιαφερόταν περισσότερο να επενδύσει πάνω τους (η Opticality Ventures), έθεσε ως όρο ότι το προϊόν που θα προέκυπτε, θα έπρεπε να είναι Ελεύθερο, διότι αλλιώς δεν βλέπανε πως θα μπορούσαν να κατακτήσουν ένα σημαντικό μερίδιο αγοράς. Η Digital Creations συμφώνησε με αυτή την προσέγγιση και λίγους μήνες αργότερα ανακοίνωσε την πρώτη έκδοση του Zope. Στις μέρες μας, η Zope Corporation εξειδικεύεται σε υπηρεσίες consulting, εκπαίδευσης και υποστήριξης για την διαχείριση συστημάτων περιεχομένου (content management systems) που βασίζονται στο Zope, και άλλα προϊόντα για τα οποία το Zope αποτελεί αναντίρρητα τον θεμελιώδη λίθο.

Η Ximian (τέως Helix Code) αποτελεί μία γνωστή περίπτωση ανάπτυξης Ελεύθερων εφαρμογών σε ένα επιχειρηματικό περιβάλλον. Στενά συνδεδεμένη, από την ίδρυσή της, με το GNOME project, η Ximian παρήγαγε εφαρμογές λογισμικού όπως το Evolution (ένας Personal Information Manager που περιλαμβάνει παρόμοια λειτουργικότητα με εκείνην του Microsoft Outlook), το Red Carpet (ένα εύχρηστο σύστημα για την διαχείριση των πακέτων ενός

λειτουργικού συστήματος) και το MONO (μία υλοποίηση ενός μεγάλου μέρους της πλατφόρμας .NET). Η εταιρεία ιδρύθηκε τον Οκτώβριο του 1999 και προσήλκυσε πολλούς developers από το GNOME, οι οποίοι και έγιναν μέλη της ομάδας ανάπτυξης (ενώ συνέχισαν, σε πολλές περιπτώσεις, να συνεργάζονται με το GNOME project). Η Ximian κατέκτησε μία θέση ως μία εταιρεία engineering με εξειδίκευση σε προσαρμογές GNOME, στη δημιουργία εφαρμογών που βασίζονται στο GNOME, και γενικά, παρέχοντας υπηρεσίες ανάπτυξης βασισμένες σε Ελεύθερο Λογισμικό, ιδιαίτερα σε εργαλεία σχετικά με το περιβάλλον της επιφάνειας εργασίας. Τον Αύγουστο του 2003, η Ximian εξαγοράστηκε από την Novell.

Η Cisco Enterprise Print System (CEPS) (<http://ceps.sourceforge.net/>) [17] είναι ένα σύστημα διαχείρισης εκτυπώσεων για οργανισμούς που μεταχειρίζονται πάρα πολλούς εκτυπωτές. Είχε αναπτυχθεί αρχικά από την Cisco, για να καλύψει τις δικές της ανάγκες και το απελευθέρωσε το 2000, κυκλοφορώντας το υπό την Άδεια GNU GPL. Είναι δύσκολο να γνωρίζεις κανείς γιατί η Cisco το έπραξε αυτό, αλλά θα μπορούσε να έχει σχέση με την ανεύρεση εξωτερικών συνεισφορών (αναφορές σφαλμάτων, νέοι controllers, patches, κλπ.). Πάντως, σε κάθε περίπτωση, εκείνο που είναι προφανές είναι ότι αφού η Cisco δεν είχε σχέδια εμπορικοποίησης του προϊόντος και η εν δυνάμει αγορά του δεν ήταν και τόσο ξεκάθαρη, είχε πολύ λίγα να χάσει με αυτή την απόφαση.

5.1.6. Άλλοι τρόποι χρηματοδότησης

Υπάρχουν και άλλοι τρόποι χρηματοδότησης, που είναι δύσκολο να τους κατατάξει κανείς στις προηγούμενες κατηγορίες. Για παράδειγμα, θα μπορούσαμε να μνημονεύσουμε τους ακόλουθους:

- Χρήση της αγοράς για να έρθουν σε επαφή οι developers με τους πελάτες. Η ιδέα πίσω από αυτή τη μορφή χρηματοδότησης, είναι ότι, ειδικά για τα μικρότερα projects, είναι δύσκολο για έναν πελάτη που επιθυμεί ένα συγκεκριμένο development, να έρθει σε επαφή με έναν developer που να είναι ικανός να το υλοποιήσει με έναν αποτελεσματικό τρόπο.

Για να βελτιωθεί αυτή η κατάσταση, οι αγορές ανάπτυξης Ελεύθερου Λογισμικού δημιουργήθηκαν εκεί όπου οι developers έχουν την δυνατότητα να διαφημίσουν τις ικανότητές τους, αλλά και οι πελάτες, να κοινοποιήσουν τι είδους development χρειάζονται. Ένας developer και ένας πελάτης συνάπτουν μία συμφωνία. Έχουμε μία κατάσταση όμοια με εκείνη που ήδη περιγράψαμε ως "χρηματοδότηση από κάποιον που αναζητά βελτιώσεις" (παράγραφος 5.1.3).

To SourceXchange

Το SourceXchange αποτελεί ένα παράδειγμα μίας αγοράς που φέρνει τους developers σε επαφή με τους εν δυνάμει πελάτες. Για να διαφημισθεί ένα project, ένας πελάτης θα κοινοποιεί ένα RFP (αίτημα για κατάθεση πρότασης) που θα διευκρινίζει το είδος του development που απαιτείται καθώς και τους πόρους που προτίθεται να συνεισφέρει για την ανάπτυξη αυτή. Αυτά τα RFPs δημοσιεύονται στο site. Όταν ένας developer εντοπίσει και που να τον ενδιαφέρει, θα υποβάλλει μία προσφορά γι' αυτό. Αν ο developer και ο πελάτης συμφωνούν με τους όρους για την ανάπτυξη, τότε θα αρχίσει ένα project. Κανονικά, το κάθε project εποπτεύεται από έναν peer reviewer, έναν επόπτη που θα είναι υπεύθυνος να διασφαλίσει ότι ο developer συμμορφώθηκε με τις προδιαγραφές και για το ότι οι προδιαγραφές, πράγματι, έχουν έννοια, συμβουλευοντας πως να συνεχισθεί το project, κλπ. Το SourceXchange (ιδιοκτησίας της εταιρείας CollabNet) ανέλαβε την ευθύνη παροχής του site, της διασφάλισης των ικανοτήτων των reviewers, εγγυώμενο την πληρωμή σε περίπτωση ολοκλήρωσης των projects και προσφέροντας εργαλεία monitoring (υπηρεσίες για τις οποίες χρέωσε τον πελάτη). Το πρώτο project με την μεσολάβηση του SourceXchange ολοκληρώθηκε τον Μάρτιο του 2000, αλλά μόλις ένα χρόνο αργότερα, τον Απρίλιο του 2001, το site έκλεισε.

- Χρηματοδότηση project μέσω πώλησης ομολόγων. Η ιδέα πίσω από αυτή τη μορφή χρηματοδότησης, είναι παρόμοια με εκείνη της κανονικής αγοράς ομολόγων που εκδίδουν οι εταιρείες, αλλά εστιασμένη στην ανάπτυξη του Ελεύθερου Λογισμικού. Υπάρχουν διάφορες παραλλαγές, αλλά μία από τις γνωστότερες λειτουργεί ως εξής. Όταν ένας developer (είτε πρόκειται για ένα άτομο, είτε πρόκειται για εταιρεία) έχει μία ιδέα για ένα πρόγραμμα, ή για μία βελτίωση ενός προγράμματος, τότε το συντάσσει σαν μία τεχνική προδιαγραφή, μαζί με μία πρόβλεψη κόστους της ανάπτυξης και, κατόπιν, εκδίδει ομόλογα με στόχο την δημιουργία του. Η αξία αυτών των ομολόγων αποκτά υπόσταση μόνο εάν το project τελικά ολοκληρωθεί. Όταν ο developer θα έχει πουλήσει αρκετά ομόλογα, τότε θα αρχίσει και η ανάπτυξη, που θα

χρηματοδοτείται με δάνεια βασισμένα σε αυτά τα ομόλογα. Όταν η ανάπτυξη θα έχει ολοκληρωθεί, και ένα ανεξάρτητο τρίτο μέρος πιστοποιήσει ότι πραγματικά περατώθηκε και ότι είναι σύμφωνη με τις τεχνικές προδιαγραφές, τότε ο developer "εκτελεί" τα ομόλογα, αποπληρώνει τα χρέη, και εφ' όσον περισσέψουν κάποια χρήματα επιπλέον, αυτά θα συνιστούν το προκύψαν κέρδος από την ανάπτυξη.

Ποιος θα μπορούσε να ενδιαφέρεται να αγοράσει αυτά τα ομόλογα; Προφανώς, εκείνοι οι χρήστες που θα θέλανε την επίτευξη του νέου προγράμματος ή την βελτίωση ενός υπάρχοντος λογισμικού. Σε κάποιο βαθμό, αυτό το σύστημα με τα ομόλογα επιτρέπει στα ενδιαφερόμενα μέρη να διαμορφώσουν τις προτεραιότητες των developers (τουλάχιστον, εν μέρει), μέσω της απόκτησης των ομολόγων. Αυτό σημαίνει, επίσης, ότι τα κόστη ανάπτυξης δεν χρειάζεται να τα επωμισθεί μόνον μία εταιρεία, αλλά μπορούν μάλλον να επιμερισθούν σε διάφορες επιχειρήσεις (περιλαμβάνοντας ακόμη και μεμονωμένα φυσικά πρόσωπα), που θα πρέπει, επιπροσθέτως, να πληρώσουν μόνον υπό την προϋπόθεση ότι το project θα ολοκληρωθεί με επιτυχία στο τέλος. Ένας μηχανισμός παρόμοιος με αυτό το σύστημα, προτείνεται, με πολλές παραπάνω λεπτομέρειες, στο "The Wall Street performer protocol. Using software completion bonds to fund open source software development", by Chris Rasch (2001) [191].

Βιβλιογραφία

Το σύστημα ομολόγων που περιγράφεται, βασίζεται στο *πρωτόκολλο του "street performer"* ("The street performer protocol", από το: *Τρίτο USENIX Workshop, στα Πρακτικά του ηλεκτρονικού εμπορίου*, 1998 [152], και το "The street performer protocol and digital copyrights", 1999 [153]), ένας μηχανισμός βασισμένος στο ηλεκτρονικό εμπόριο, που σχεδιάστηκε για να διευκολύνει την ιδιωτική χρηματοδότηση Ελεύθερων έργων. Εν συντομία, όποιος και αν ενδιαφέρεται για ένα συγκεκριμένο έργο, θα πρέπει να υποσχεθεί επισήμως ότι θα πληρώσει ένα κάποιο ποσό, εφ' όσον το έργο ολοκληρωθεί και δημοσιευθεί ως Ελεύθερο. Ο στόχος είναι να βρεθεί ένας νέος τρόπος χρηματοδότησης των σχετικά μικρών έργων, που μπορούν να διατεθούν προς τον καθένα. Η μέθοδος αυτή, όμως, μπορεί να επεκταθεί κατά πολλούς άλλους τρόπους (ένας από τους οποίους μπορεί να είναι και τα ομόλογα για την δημιουργία Ελεύθερου Λογισμικού). Θα λέγαμε ότι υπάρχει χώρος για να τεθεί σε πρακτική εφαρμογή ένα παράγωγο αυτού του πρωτοκόλλου, το *ορθολογικό πρωτόκολλο του street performer* (Paul Harrison, 2002, [137]) όπου εφαρμόζεται το http://www.csse.monash.edu.au/~pfh/circle/funding_results.html για την συλλογή κεφαλαίων, προς χρηματοδότηση του "The Circle", ενός project Ελεύθερου Λογισμικού.

- **Συνεταιρισμοί για Developers.** Σε αυτή την περίπτωση, οι developers Ελεύθερου Λογισμικού, αντί να δουλεύουν είτε ατομικά, είτε για κάποια εταιρεία, συνασπίζονται για τη δημιουργία μίας μορφής Ένωσης (που θα μοιάζει, κανονικά, με έναν Συνεταιρισμό). Από όλες τις άλλες απόψεις, θα λειτουργεί με τον ίδιο τρόπο σαν μία εταιρεία, με μία έμφαση στην ηθική της δέσμευση στο Ελεύθερο Λογισμικό, που μπορεί να αποτελεί και μέρος του Καταστατικού της επιχείρησης (αν και μία κανονική εταιρεία μπορεί, επίσης, να το πράξει). Σε αυτό το είδος οργάνωσης, μπορούμε να διακρίνουμε μία ποικιλία συνδυασμών τόσο εθελοντικών όσο και αμοιβόμενων μορφών εργασίας. Ένα παράδειγμα αποτελούν οι Free Developers.

- **Σύστημα Δωρεών.** Αυτό συνεπάγεται την ενεργοποίηση ενός μηχανισμού για την αμοιβή του δημιουργού ενός συγκεκριμένου λογισμικού, μέσω της ιστοσελίδας που φιλοξενεί το project. Κατ' αυτόν τον τρόπο, οι χρήστες που ενδιαφέρονται για ένα project που συνεχίζει να παράγει νέες εκδόσεις, μπορούν να το υποστηρίξουν οικονομικά μέσω εθελοντικών δωρεών, σαν μία μορφή χρηματοδότησης για τον developer.

5.2. Επιχειρηματικά μοντέλα που βασίζονται στο Ελεύθερο Λογισμικό

Πέραν των μηχανισμών χρηματοδότησης ενός project, για τους οποίους ήδη συζητήσαμε, υπάρχει και μία άλλη πλευρά της οικονομίας που αξίζει ιδιαίτερης αναφοράς και πρόκειται για τα επιχειρηματικά μοντέλα. Όταν μιλήσαμε για τους μηχανισμούς χρηματοδότησης, αναφέραμε ήδη

μερικά τέτοια μοντέλα, ακροθιγώς. Σε αυτό εδώ το κεφάλαιο, θα τα περιγράψουμε με έναν πιο μεθοδικό τρόπο.

Σε γενικές γραμμές, μπορούμε αν πούμε ότι είναι υπό διερεύνηση πολλά επιχειρηματικά μοντέλα που περιστρέφονται γύρω από το Ελεύθερο Λογισμικό. Μερικά κάπως πιο κλασικά, ενώ άλλα είναι πιο καινοτόμα. Πρέπει να λάβουμε υπ' όψη μας ότι δεν είναι εύκολο να χρησιμοποιήσουμε τα μοντέλα που βασίζονται στις πωλήσεις Αδειών χρήσεως, που είναι και τα πιο διαδεδομένα στην βιομηχανία του λογισμικού, αφού είναι πολύ δύσκολη η εκμετάλλευση αυτού του συγκεκριμένου μηχανισμού μέσα στον κόσμο του Ελεύθερου Λογισμικού. Μπορούμε, πάντως, να χρησιμοποιήσουμε τα μοντέλα που βασίζονται σε υπηρεσίες προς τρίτους, με το πλεονέκτημα ότι είναι δυνατό να προσφέρει κανείς την πλήρη υποστήριξη του προγράμματος, χωρίς να είναι απαραίτητος ο ίδιος ο δημιουργός του.

Πώληση Ελεύθερου Λογισμικού έναντι ενός ποσού Χ ανά αντίτυπο

Στον κόσμο του Ελεύθερου Λογισμικού είναι δύσκολο να χρεώνει κανείς για Άδειες χρήσεως, αλλά όχι και πλήρως αδύνατον. Σε γενικές γραμμές, δεν υπάρχει τίποτε στους ορισμούς του Ελεύθερου Λογισμικού που να εμποδίζει μία εταιρεία να δημιουργήσει ένα προϊόν και να το διανέμει μόνον σε εκείνους που είναι διατεθειμένοι να πληρώσουν ένα κάποιο ποσό. Για παράδειγμα, ένας κάποιος παραγωγός θα μπορούσε να αποφασίσει να διανείμει το προϊόν του με μία Ελεύθερη Άδεια, αλλά μόνον σε όποιον πληρώνει 1,000 ευρώ ανά αντίτυπο (όπως και στον κλασικό κόσμο του ιδιοταγούς λογισμικού).

Πάντως, αν και θα ήταν θεωρητικώς εφικτό, είναι πρακτικώς δύσκολο να συμβεί και τέτοιο. Διότι, άπαξ και ο παραγωγός πουλήσει το πρώτο αντίτυπο, τότε οποιοσδήποτε αποδέκτης μπορεί να διακρίνει ένα κίνητρο για να προσπαθήσει να ανακτήσει την καταβληθείσα επένδυσή του, πουλώντας περαιτέρω αντίτυπα σε μία χαμηλότερη τιμή (και που δεν μπορεί να απαγορευθεί από την Άδεια του προγράμματος, εφ' όσον αυτή είναι Ελεύθερη). Στο προηγούμενο παράδειγμα, αν ένας προσπαθήσει να πουλήσει 10 αντίτυπα, έναντι 100 ευρώ το καθένα, θα σήμαινε ότι, μεταξύ των άλλων, το προϊόν θα προέκυπτε ουσιαστικά δωρεάν (αλλά επίσης, θα καθιστούσε πολύ δύσκολο για τον αρχικό παραγωγό να πουλήσει ακόμη ένα αντίτυπο έναντι 1000 ευρώ, αφού αυτό το προϊόν μπορεί να αγορασθεί, παράνομα, έναντι 1/10 της κανονικής τιμής). Είναι εύκολο να προβλέψουμε πως αυτή η διαδικασία θα μπορούσε εύκολα να εξελιχθεί σε ένα καταρράκτη, μέχρι όπου να πωλούνται αντίτυπα σε μία τιμή που θα είναι κοντά στο οριακό κόστος της απλής αντιγραφής, πράγμα που, υπό τις σημερινές συνθήκες, ισοδυναμεί, πρακτικά, με μηδέν.

Ακόμη κι' έτσι, και κρατώντας κατά νου ότι ο περιγραφείς μηχανισμός σημαίνει ότι ένας παραγωγός δεν μπορεί, κανονικά, να βάλει μία τιμή (και ιδιαίτερα μία υψηλή τιμή) στην απλή αναδιανομή του προγράμματος, υπάρχουν επιχειρηματικά μοντέλα που προβλέπουν ακριβώς αυτό. Ένα παράδειγμα είναι οι διανομές GNU/Linux, που πωλούνται σε πολύ χαμηλότερες τιμές, σε σύγκριση με τις αντίστοιχες ιδιοταγείς ανταγωνιστικές λύσεις, αλλά σίγουρα παραπάνω (και συνήθως πολύ παραπάνω) από το κόστος του απλού αντιγράφου (έστω και μπορεί κάποιος να το κατεβάσει ελεύθερα από το Διαδίκτυο). Σε αυτές τις περιπτώσεις, υπεισέρχονται, μετά βεβαιότητας, άλλοι παράγοντες, όπως η εταιρική εικόνα ή ο βαθμός ευκολίας για τον καταναλωτή. Αλλά αυτή δεν αποτελεί και την μόνη περίπτωση. Συνεπώς, αντί να λέγεται ότι το Ελεύθερο Λογισμικό “δεν μπορεί να πωλείται έναντι ενός ποσού Χ ανά αντίτυπο”, θα πρέπει να θυμόμαστε ότι είναι πολύ δύσκολο να γίνει κάτι τέτοιο, και ότι μπορεί, πιθανώς, να δημιουργεί μικρότερο κέρδος, αλλά υπάρχουν μοντέλα που βασίζονται ακριβώς σε αυτό.

Με δεδομένους αυτούς τους περιορισμούς (και αυτά τα πλεονεκτήματα), και ενώ δοκιμάζονται εδώ και αρκετά χρόνια διάφορες παραλλαγές των συνηθισμένων επιχειρηματικών μοντέλων της βιομηχανίας του λογισμικού, συγχρόνως αναζητούνται και άλλες καινοτόμες λύσεις για την εκμετάλλευση των δυνατοτήτων που προσφέρει το Ελεύθερο Λογισμικό. Χωρίς αμφιβολία, μέσα στα επόμενα λίγα χρόνια, θα είμαστε μάρτυρες ακόμη περισσότερου πειραματισμού σε αυτόν τον τομέα, και θα έχουμε επίσης και περισσότερη πληροφόρηση σχετικά με το ποια μοντέλα μπορούν να λειτουργήσουν κάτω από αυτές τις συνθήκες.

Σε αυτό το κεφάλαιο προσφέρουμε μία πανοραμική άποψη των επιχειρηματικών μοντέλων που συναντώνται συχνότερα σήμερα, κατατάσσοντάς τα ανά ομάδες, με την πρόθεση να δείξουμε στον αναγνώστη τα κοινά στοιχεία που έχουν μεταξύ τους και εκείνα που τα διαφοροποιούν, εστιάζοντας

σε εκείνα που προσφέρουν ανάπτυξη προϊόντων και υπηρεσιών γύρω από ένα προϊόν Ελεύθερου Λογισμικού. Η δημιουργία εσόδων, σε αυτή την περίπτωση, προέρχεται άμεσα από τις δραστηριότητες ανάπτυξης και τις υπηρεσίες γι' αυτό το προϊόν, χωρίς αυτό να σημαίνει απαραίτητως την ανάπτυξη ενός νέου προϊόντος. Όταν μία τέτοια ανάπτυξη λάβει χώρα, αυτά τα μοντέλα προβλέπουν την χρηματοδότηση των προϊόντων Ελεύθερου Λογισμικού ως ένα υποπροϊόν, με την έννοια ότι πρόκειται για ιδιαιτέρως ενδιαφέροντα μοντέλα, με μία εν δυνάμει τεράστια επίδραση στον κόσμο του Ελεύθερου Λογισμικού γενικά.

Σε κάθε περίπτωση, και παρ' ότι εμείς εδώ παραθέτουμε μία σχετικά ξεκάθαρη κατάταξη, δεν πρέπει να ξεχνάμε ότι σχεδόν όλες οι εταιρείες, στην πραγματικότητα, χρησιμοποιούν συνδυασμούς αυτών των μοντέλων που εμείς περιγράφουμε, με άλλα πιο παραδοσιακά.

5.2.1. Καλύτερη γνώση

Η εταιρεία που ακολουθεί αυτό το επιχειρηματικό μοντέλο, προσπαθεί να δημιουργήσει κέρδη από την γνώση της για ένα ελεύθερο προϊόν (ή ένα σύνολο προϊόντων). Τα έσοδά της θα προέλθουν από πελάτες προς τους οποίους παρέχει ή πουλάει, ήδη, υπηρεσίες που σχετίζονται με αυτή τη γνώση: ανάπτυξη που να βασίζεται πάνω στο προϊόν, τροποποίηση, προσαρμογή, εγκατάσταση και ενσωμάτωση με άλλα προϊόντα. Το ανταγωνιστικό πλεονέκτημα της εταιρείας θα έχει άμεση σχέση με την καλύτερη τεχνογνωσία της για το προϊόν: συνεπώς, η εταιρεία θα βρίσκεται σε μία ιδιαίτερα προνομιακή θέση εάν είναι συγχρόνως και ο παραγωγός ή ένας δραστήριος συμμετέχων στο project που προωθεί το συγκεκριμένο προϊόν λογισμικού.

Αυτός είναι ένας από τους λόγους που διάφορες εταιρείες που χρησιμοποιούν αυτό το μοντέλο τείνουν να γίνονται ενεργοί συμμετοχοί στα projects τα οποία παράγουν το λογισμικό, γύρω από το οποίο εκείνες προσπαθούν να πωλήσουν υπηρεσίες: αυτός είναι ένας πολύ αποτελεσματικός τρόπος για να κερδίσουν σε γνώση σχετικά με αυτό, και ακόμη πιο σημαντικό, για να μπορεί να αναγνωρισθεί αυτή η τεχνογνωσία. Οποσδήποτε, το να είσαι σε θέση να μπορείς να πεις σε έναν πελάτη ότι στο προσωπικό της εταιρείας συγκαταλέγονται διάφοροι developers από το ίδιο το project που παράγει το λογισμικό, που, επί παραδείγματι, χρειάζεται τροποποίηση, αντιπροσωπεύει σαφώς μία καλή εγγύηση.

Σχέση με τα projects ανάπτυξης

Συνεπώς, οι εταιρείες αυτού του είδους ενδιαφέρονται ιδιαίτερα να αναδίδουν την εικόνα κάποιου που διαθέτει καλή γνώση πάνω σε συγκεκριμένα Ελεύθερα προϊόντα. Άρα, ένα ενδιαφέρον μεθεπόμενο είναι ότι η υποστήριξη για τα projects Ελεύθερου Λογισμικού (για παράδειγμα, συμμετέχοντας ενεργώς σε αυτά, ή επιτρέποντας στους υπαλλήλους να το πράξουν κατά τη διάρκεια της εργάσιμης ημέρας) δεν συνιστά κάτι καθαρά φιλανθρωπικό. Αντιθέτως, μπορεί να αποτελέσει ένα από τα πιο κερδοφόρα χαρτιά μίας εταιρείας, αφού οι πελάτες θα εκτιμήσουν πολύ θετικά, ως ένα ξεκάθαρο σήμα, ότι η εταιρεία κατέχει αληθινή τεχνογνωσία σχετικά με το εν λόγω προϊόν. Κατ' αυτό τον τρόπο, θα είναι επιπλέον σε θέση να παρακολουθεί στενά την ανάπτυξή του, προσπαθώντας να διασφαλίσει, για παράδειγμα, ότι οι βελτιώσεις που επιθυμούν οι πελάτες της θα ενσωματωθούν στο προϊόν που αναπτύσσεται από το project.

Αν αυτό το αναλύσουμε από μία πιο γενική σκοπιά, θα δούμε ότι πρόκειται για μία κατάσταση στην οποία αμφότερα τα μέρη, δηλαδή τόσο η εταιρεία όσο και το project, επωφελούνται από την συνεργασία. Το μεν project επωφελείται, είτε από την ανάπτυξη που η εταιρεία καθιστά δυνατή, είτε διότι μερικοί από τους developers αμείβονται (τουλάχιστον σε μία βάση μερικής απασχόλησης) για την εργασία τους στο project. Η δε εταιρεία, επωφελείται σε γνώση γύρω από το προϊόν, σε εικόνα προς τους πελάτες της, και στο βαθμό που μπορεί να επηρεάζει το όλο project.

Το φάσμα των υπηρεσιών που παρέχουν αυτού του είδους οι εταιρείες, μπορεί να είναι ευρύτατο, αλλά, συνήθως, συνίσταται σε παραμετροποιημένες επεκτάσεις, προσαρμογές ή ενσωματώσεις των προϊόντων πάνω στα οποία υπάρχει ειδημοσύνη, ή σε παροχή υπηρεσιών consulting, όπου

συμβουλεύουν τους πελάτες τους πως να χρησιμοποιούν καλύτερα το εν λόγω προϊόν (ειδικά όταν πρόκειται για ένα σύνθετο προϊόν, ή εφ' όσον η ορθή λειτουργία του είναι κρίσιμης σημασίας για τον πελάτη).

Παραδείγματα

Παραδείγματα εταιρειών που χρησιμοποίησαν αυτό το επιχειρηματικό μοντέλο, μέχρι ενός ορισμένου βαθμού, περιλαμβάνουν τις παρακάτω επιχειρήσεις:

- LinuxCare (<http://www.linuxcare.com>) [45]. Ιδρύθηκε το 1996, αρχικά παρείχε συμβουλευτικές υπηρεσίες και υποστήριξη για το GNU/Linux και το Ελεύθερο Λογισμικό στις ΗΠΑ, και το προσωπικό της απελευθεύει ουσιαστικά από ειδικούς στο GNU/Linux. Σε κάθε περίπτωση, το 2002 οι στόχοι της άλλαξαν και έκτοτε εξειδικεύθηκε στην παροχή υπηρεσιών σχεδόν κατ' αποκλειστικότητα στο GNU/Linux, που τρέχει σε εικονικές μηχανές, σε μεγάλους υπολογιστές της IBM. Άλλαξε, επίσης, και το επιχειρηματικό της μοντέλο "better knowledge with limitations", αφού, σαν ένα βασικό μέρος των υπηρεσιών της, προσφέρει μία μη-ελεύθερη εφαρμογή, τη Levanta.
- Alcôve (<http://www.alcove.com>) [3]. Ιδρύθηκε το 1997 στη Γαλλία και προσφέρει κυρίως υπηρεσίες συμβουλευτικής στο Ελεύθερο Λογισμικό, συμβουλευτική σε θέματα στρατηγικής, υποστήριξης και ανάπτυξης. Από την ίδρυσή της, η Alcôve κράτησε ως προσωπικό της τους developers διαφόρων Ελεύθερων projects, προσπαθώντας να επιτύχει μία απόδοση υπό μορφή της βελτιωμένης εικόνας της. Προσπάθησε, επίσης, να προβάλει μία εικόνα μίας εταιρείας που, σε γενικές γραμμές, είναι συνδεδεμένη με την Κοινότητα του Ελεύθερου Λογισμικού, συνεργαζόμενη, για παράδειγμα, με Συλλόγους χρηστών και διαφημίζοντας τις συνεργασίες της με τα διάφορα Ελεύθερα projects (π.χ. μέσω των Alcôve-Labs [4]).

5.2.2. Καλύτερη γνώση με περιορισμούς

Αυτά τα μοντέλα μοιάζουν με εκείνα που περιγράψαμε στο προηγούμενο κεφάλαιο, αλλά προσπαθούν να περιορίσουν τον ανταγωνισμό που ενδεχομένως θα έχουν να αντιμετωπίσουν. Ενώ στα καθαρά μοντέλα που βασίζονται στην καλύτερη γνώση, ο οποιοσδήποτε μπορεί, επί της αρχής, να συμμετάσχει στον ανταγωνισμό, αφού το χρησιμοποιούμενο λογισμικό είναι το ίδιο (και Ελεύθερο), σε αυτή την περίπτωση η προσπάθεια είναι να αποφευχθεί αυτή η κατάσταση μέσω της έγερσης εμποδίων στον ανταγωνισμό. Αυτά τα εμπόδια προσλαμβάνουν συνήθως τη μορφή διπλωμάτων ευρεσιτεχνίας (patents) ή ιδιοταγών (proprietary) αδειών, που επηρεάζουν κανονικά ένα μικρό (πλην ουσιώδους σημασίας) κομμάτι του υπό ανάπτυξη προϊόντος.

Αυτός είναι ο λόγος για τον οποίο αυτά τα μοντέλα μπορούν να θεωρηθούν ως μεικτά, υπό την έννοια ότι βρίσκονται κάπου ανάμεσα στο Ελεύθερο και το ιδιοταγές λογισμικό.

Σε πολλές περιπτώσεις, η κοινότητα Ελεύθερου Λογισμικού αναπτύσσει τη δική της έκδοση, πράγμα που σημαίνει ότι το ανταγωνιστικό πλεονέκτημα μπορεί να εξαφανισθεί, ή ακόμη και να αποβεί εις βάρος της εν λόγω εταιρείας, εάν ο Ελεύθερος ανταγωνιστής αναδειχθεί ως ηγέτης της αγοράς και οι πελάτες της εταιρείας στρέψουν την ζήτησή τους σε αυτό.

Παραδείγματα

Υπάρχουν πολλές περιπτώσεις που χρησιμοποιούν αυτό το επιχειρηματικό μοντέλο, αφού θεωρείται συχνά ως λιγότερο επικίνδυνο από εκείνο της καθαρής γνώσης. Σε κάθε περίπτωση, οι εταιρείες που το χρησιμοποίησαν, εξελίχθηκαν κατά διαφόρους τρόπους. Μερικοί από αυτούς περιλαμβάνουν:

- Caldera (<http://www.sco.com>) [16]. Η ιστορία της Caldera είναι περίπλοκη. Στην αρχή, δημιούργησε τη δική της διανομή GNU/Linux, με στόχο τις επιχειρήσεις: την Caldera OpenLinux. Το 2001 εξαγόρασε τον τομέα Unix από την SCO, και το 2002 άλλαξε το όνομά της σε SCO Group. Η επιχειρηματική της στρατηγική άλλαξε εξ' ίσου συχνά όσο και το όνομά της, από την ολική της υποστήριξη προς το GNU/Linux, μέχρι τις νομικές της διαμάχες με την IBM και την Red Hat στο 2003, εγκαταλείποντας τη δική της διανομή. Αλλά, σχετικά με αυτό, ο τρόπος του επιχειρείν της Caldera, τουλάχιστον μέχρι το 2002, συνιστά ένα καθαρό μοντέλο καλύτερης γνώσης με περιορισμούς. Η Caldera προσπάθησε να εκμεταλλευθεί την γνώση της για την πλατφόρμα GNU/Linux, αλλά περιορίζοντας τον ανταγωνισμό

που θα μπορούσε να αντιμετωπίσει, μέσω της εισαγωγής ιδιοταγούς λογισμικού στην διανομή της. Αυτό καθιστά δύσκολο για τους πελάτες της να αλλάξουν διανομή, άπαξ και την χρησιμοποιήσουν, διότι αν και οι άλλες διανομές GNU/Linux περιελάμβαναν το ελεύθερο κομμάτι της Caldera OpenLinux, ωστόσο, δεν ενσωμάτωναν το ιδιοταγές κομμάτι.

• **Ximian** (<http://ximian.com/>) [74]. Ιδρύθηκε το 1999 υπό το όνομα Helix Code, από developers στενά συνδεδεμένους με το GNOME project, εξαγοράστηκε τον Αύγουστο του 2003 από την Novell. Το πλείστο μέρος λογισμικού που ανεπτύχθη ήταν Ελεύθερο (σε γενικές γραμμές, τμήμα του GNOME). Πάντως, σε ένα πολύ συγκεκριμένο τμήμα, η Ximian αποφάσισε να αδειοδοτήσει ένα τμήμα ως ιδιοταγές λογισμικό: το Connector for Exchange. Αυτό το κομμάτι επιτρέπει σε ένα από τα κορυφαία προϊόντα της, στο Evolution (έναν Personal Information Manager που περιλαμβάνει δυνατότητες e-mail, ατζέντας, ημερολογίου, κλπ), να μπορεί να επικοινωνεί με τους διακομιστές του Microsoft Exchange, που χρησιμοποιούνται συχνά από τις μεγάλες εταιρείες. Αυτός είναι και ο τρόπος με τον οποίο προσπάθησε να ανταγωνισθεί με ένα πλεονέκτημα επί των άλλων εταιρειών που προσέφεραν υπηρεσίες βασισμένες στο GNOME, με τα προϊόντα που ανέπτυξε η ίδια η Ximian, αλλά που δεν μπορούσαν να επικοινωνήσουν και τόσο εύκολα με το Exchange. Με την εξαίρεση αυτού του προϊόντος, το μοντέλο της Ximian ήταν εκείνο της "καλύτερης γνώσης", και που, επίσης, βασίστηκε στο ότι ήταν η πηγή του προγράμματος (όπως και θα δούμε και αργότερα). Σε κάθε περίπτωση, αυτό το κομμάτι κυκλοφόρησε ως Ελεύθερο Λογισμικό κατά το 2005.

5.2.3. Πηγή ενός προϊόντος Ελεύθερου Λογισμικού

Αυτό το μοντέλο μοιάζει με εκείνο που βασίζεται στην καλύτερη γνώση, αλλά με μία εξειδίκευση, υπό την έννοια ότι η εταιρεία που το χρησιμοποιεί είναι ο παραγωγός, σχεδόν εξ' ολοκλήρου, ενός ελεύθερου προϊόντος. Φυσικά, με το να είσαι ο παραγωγός του εν λόγω προϊόντος, το ανταγωνιστικό πλεονέκτημα αυξάνει, ελέγχοντας την εξέλιξή του και κρατώντας το μπροστά από τον ανταγωνισμό. Όλα αυτά τοποθετούν την εταιρεία που το αναπτύσσει σε πολύ ισχυρή θέση προς τους πελάτες που αναζητούν υπηρεσίες για αυτό το πρόγραμμα.

Αποτελεί, επίσης, ένα πολύ ενδιαφέρον μοντέλο, από την άποψη της προς τα έξω εικόνας, αφού η εταιρεία αποδεικνύει το δυναμικό της για ανάπτυξη με την δημιουργία και τη συντήρηση της εν λόγω εφαρμογής, που μπορεί να είναι κάτι πολύ χρήσιμο όταν πρόκειται να πείσει τους πελάτες σχετικά με τις ικανότητές της. Με ανάλογο τρόπο, χτίζει μία καλή εικόνα, γενικά και προς την κοινότητα του Ελεύθερου Λογισμικού, αφού αυτή γίνεται αποδέκτης ενός νέου ελεύθερου προϊόντος από την εταιρεία, που το μετατρέπεται και αυτό σε ένα κομμάτι δημοσίων προσβάσιμο.

Παραδείγματα

Πολλά από τα ελεύθερα προϊόντα άρχισαν να αναπτύσσονται στο πλαίσιο μιας εταιρείας και, πολύ συχνά, αυτή η εταιρεία συνέχισε να καθοδηγεί την μετέπειτα εξέλιξή του. Μερικά παραδείγματα:

• **H Ximian**. Αναφέραμε ήδη πως χρησιμοποίησε το μοντέλο της καλύτερης γνώσης με περιορισμούς. Αλλά, σε γενικές γραμμές, η Ximian ακολούθησε το καθαρό μοντέλο που βασίζεται στο να είσαι η πηγή των ελεύθερων προγραμμάτων. Τα βασικά προϊόντα της, σαν το Evolution ή το Red Carpet, διανεμήθηκαν υπό Αδειες GPL. Πάντως, και άλλα σημαντικά προϊόντα σαν την Mono, διανέμονται επί το πλείστον με Αδειες MIT X11 ή LGPL. Σε κάθε περίπτωση, η Ximian ανέπτυξε τα δικά της προϊόντα σχεδόν αποκλειστικά από την αρχή. Η εταιρεία προσπάθησε να πετύχει μία απόδοση για αυτές τις προσπάθειες ανάπτυξης, επιδιώκοντας συμφωνίες για να τα πιλοτάρει κατά συγκεκριμένους τρόπους, προσαρμόζοντάς τα στις ανάγκες των πελατών και προσφέροντας παραμετροποίηση και συντήρηση.

• **H Zope Corporation** (<http://www.zope.com/>) [75]. Το 1995, ιδρύθηκε η Digital Creations, αναπτύσσοντας ένα ιδιοταγές προϊόν για την διαχείριση των διαφημίσεων στο web. Το 1997, έγινε αποδέκτης μίας ένεσης κεφαλαίων, εκ μέρους, μεταξύ άλλων, μίας εταιρείας επενδυτικών κεφαλαίων (venture capital) με το όνομα Opticality Ventures. Το περίεργο σχετικά με αυτή την επένδυση (σε εκείνη την χρονική περίοδο) ήταν οι όροι που επεβλήθησαν κατά την διανομή του ανεπτυγμένου προϊόντος υπό την μορφή του Ελεύθερου Λογισμικού, που αργότερα μετετράπη σε Zope, ένας από τους πλέον δημοφιλείς διαχειριστές περιεχομένου στο Διαδίκτυο. Έκτοτε, το επιχειρηματικό μοντέλο της εταιρείας ήταν να παραγάγει το Zope και άλλα σχετικά προϊόντα, και να προσφέρει υπηρεσίες προσαρμογής και συντήρησης για όλα αυτά. Η Zope Corporation κατάφερε, επίσης, να δημιουργήσει μία δυναμική κοινότητα developers Ελεύθερου Λογισμικού γύρω από τα προϊόντα της, καθώς και να συνεργάζεται ενεργώς μαζί τους.

5.2.4. Προϊόν με πηγαίο κώδικα που έχει περιορισμούς

Αυτό το μοντέλο μοιάζει με το προηγούμενο, αλλά λαμβάνει μέτρα για να περιορίσει τον ανταγωνισμό ή να μεγιστοποιήσει τα έσοδα. Ανάμεσα στους πιο κοινούς περιορισμούς, μπορούμε να συναντήσουμε τα εξής:

- **Ιδιοταγής διανομή για ένα διάστημα, και κατόπιν, κυκλοφορία ως Ελεύθερο Λογισμικό.** Με ή χωρίς την υπόσχεση για ελεύθερη διανομή σε δεύτερο χρόνο, κάθε νέα έκδοση του προϊόντος πωλείται ως ιδιοταγές λογισμικό. Μετά από κάποια χρονική περίοδο (συνήθως, όταν μία νέα έκδοση κυκλοφορεί, επίσης με την μορφή του ιδιοταγούς λογισμικού), η παλιά έκδοση διανέμεται με μία Ελεύθερη Άδεια. Κατ' αυτό τον τρόπο, η εταιρεία παραγωγής προσπορίζεται έσοδα από πελάτες που ενδιαφέρονται να έχουν τις νέες εκδόσεις, και συγχρόνως περιορίζει τον ανταγωνισμό, αφού οποιαδήποτε άλλη εταιρεία που θέλει να ανταγωνισθεί την χρήση αυτού του προϊόντος, μπορεί να το πράξει με την ελεύθερη έκδοση (η οποία διατίθεται μόνον όταν η νέα ιδιοταγής έκδοση κυκλοφορεί, που είναι - κατά τεκμήριο- πιο βελτιωμένη και πιο πλήρης).

- **Περιορισμένη διανομή για μία περίοδο.** Σε αυτή την περίπτωση, το λογισμικό διατίθεται ως Ελεύθερο, όπως κατά την στιγμή που πρωτοκυκλοφόρησε. Αλλά, καθώς δεν υπάρχει τίποτε στην Ελεύθερη Άδεια που να υποχρεώνει την διανομή του προγράμματος προς κάθε έναν που το θέλει (αυτό είναι κάτι που το πρόσωπο που κατέχει το πρόγραμμα μπορεί να επιλέξει να κάνει ή όχι), ο παραγωγός το διανέμει, για ένα κάποιο διάστημα, προς τους πελάτες του μόνον, οι οποίοι πληρώνουν για αυτό (συνήθως με την μορφή ενός συμβολαίου υποστήριξης). Μετά από λίγο, το διανέμει προς τον καθένα, θέτοντάς το, για παράδειγμα, σε ένα δημόσιο αποθετήριο. Με αυτό τον τρόπο, ο παραγωγός κερδίζει έσοδα από τους πελάτες του, οι οποίοι αντιλαμβάνονται αυτή την κατά προτεραιότητα διαθεσιμότητα του λογισμικού, σαν μία προστιθέμενη αξία. Φυσικά, το μοντέλο δουλεύει μόνον εφ' όσον οι πελάτες δεν διανέμουν, με τη σειρά τους, δημοσίως το πρόγραμμα, άπαξ και το αποκτήσουν. Για ορισμένους τύπους πελατών, αυτό δεν είναι συχνό.

Γενικά, οι εταιρείες ανάπτυξης, σε αυτές τις περιπτώσεις, προσπορίζονται αυτά τα προαναφερθέντα οφέλη, αλλά όχι με μηδενικό κόστος. Λόγω της καθυστέρησης με την οποία διατίθεται το προϊόν προς την κοινότητα του Ελεύθερου Λογισμικού, καθίσταται για αυτήν πρακτικώς αδύνατον να συνεισφέρει στην ανάπτυξή του, με την έννοια ότι ο παραγωγός θα επωφεληθεί πολύ λίγο από εξωτερικές συνεισφορές.

Παραδείγματα

Μερικές εταιρείες που χρησιμοποιούν αυτό το επιχειρηματικό μοντέλο είναι οι εξής:

- Η Artofcode LLC (<http://artofcode.com/>) [9]. Από το έτος 2000, η Artofcode πωλεί το Ghostscript σε τρεις εκδόσεις (σε προηγούμενες εποχές, αυτό το είχε κάνει με ένα παρόμοιο μοντέλο η Alladin Enterprises). Η πιο πρόσφατη έκδοση διανέμεται ως AFPL Ghostscript, υπό μία ιδιοταγή άδεια (η οποία επιτρέπει την χρήση και την μη-εμπορική διανομή). Η επόμενη (με μία καθυστέρηση, ενός περίπου, χρόνου) διανέμεται ως GNU Ghostscript, υπό την GNU GPL. Για παράδειγμα, το καλοκαίρι του 2003, η έκδοση AFPL είναι η 8.11 (διανεμήθη στις 16 Αυγούστου), ενώ η έκδοση GNU είναι η 7.07 (που διανεμήθη αυτή καθεαυτή στις 17 Μαΐου, αλλά η ισοδύναμη με αυτήν AFPL- έκδοση ανάγεται στο 2002). Επίσης, η Artofcode προσφέρει μία τρίτη έκδοση, με μία ιδιοταγή άδεια που επιτρέπει την ενσωμάτωσή της με προϊόντα που ασύμβατα με την GNU GPL (σε αυτή την περίπτωση, χρησιμοποιεί ένα διττό μοντέλο, που θα περιγράψουμε στη συνέχεια).

- Η Ada Core Technologies (<http://www.gnat.com/>) [2]. Ιδρύθηκε το 1994 από τους δημιουργούς του πρώτου μεταγλωττιστή Ada 95, αναπτύχθηκε με μερική χρηματοδότηση από την κυβέρνηση των ΗΠΑ, βασισμένη στο GCC, τον μεταγλωττιστή GNU. Από την αρχή, τα προϊόντα της κυκλοφόρησαν ως Ελεύθερο Λογισμικό. Αλλά τα περισσότερα από αυτά προσφέρθηκαν, κατ' αρχάς, στους ίδιους τους πελάτες της, ως τμήμα ενός συμβολαίου υποστήριξης. Ο μεταγλωττιστής της, για παράδειγμα, ο οποίος συνεχίζει να βασίζεται στον GCC και διανέμεται υπό

την GNU GPL, προσφέρεται στους πελάτες της ως GNAT Pro. Η εταιρεία Ada Core Technologies δεν προσφέρει αυτό τον μεταγλωττιστή στο ευρύ κοινό με οποιονδήποτε τρόπο, και δεν μπορεί κανείς, με κανονικό τρόπο, να βρει διάφορες εκδόσεις του στο Διαδίκτυο. Εν πάσει περιπτώσει, με μία ποικίλη καθυστέρηση (περίπου ενός έτους), η Ada Core Technologies προσφέρει στο κοινό τις διάφορες εκδόσεις του δικού της μεταγλωττιστή, που είναι πολύ παρόμοιος αλλά στερείται της οποιαδήποτε υποστήριξης, υπό μορφή ενός αρχείου FTP διαθέσιμου με ανώνυμη πρόσβαση.

5.2.5. Ειδικές άδειες

Στα πλαίσια αυτών των μοντέλων, η εταιρεία παράγει ένα προϊόν το οποίο διακινεί υπό δύο ή και περισσότερες άδειες. Η μία τουλάχιστον εξ αυτών είναι Ελεύθερο Λογισμικό, αλλά οι υπόλοιπες είναι τυπικά ιδιοταγείς και επιτρέπουν στο προϊόν να πωλείται με έναν, κατά το μάλλον ή ήττον, παραδοσιακό τρόπο. Συνήθως, αυτές οι πωλήσεις συμπληρώνονται με την πώληση συμβουλευτικών υπηρεσιών και παραγώγων που συσχετίζονται με το προϊόν. Μία εταιρεία μπορεί, για παράδειγμα, να διανέμει ένα προϊόν ως Ελεύθερο Λογισμικό υπό την GNU GPL, αλλά να προσφέρει και μία ιδιοταγή έκδοση (συγχρόνως και χωρίς καθυστέρηση ανάμεσα στις 2 εκδόσεις) προς εκείνους που δεν επιθυμούν τους όρους της GPL, για παράδειγμα, διότι επιθυμούν να ενσωματώσουν το προϊόν αυτό, με το αντίστοιχο ιδιοταγές (πράγμα που η GPL δεν επιτρέπει).

Παράδειγμα

Το λογισμικό Sleepycat Software (<http://www.sleepycat.com/download/oslicense.html>) [60]. Αυτή η εταιρεία ιδρύθηκε το 1996 και δήλωσε ότι πραγματοποίησε ένα κέρδος από την αρχή (πράγμα που είναι, ασφαλώς, αξιοσημείωτο για μία εταιρεία που σχετίζεται με το λογισμικό). Τα προϊόντα της, περιλαμβανομένου του Berkeley DB (ενός πολύ δημοφιλούς Διαχειριστή Δεδομένων, αφού μπορεί εύκολα να ενσωματωθεί σε άλλες εφαρμογές), διανέμονται υπό μία ελεύθερη άδεια που διευκρινίζει ότι σε περίπτωση ενσωμάτωσης με ένα άλλο προϊόν, τότε θα πρέπει να αποδώσει τον πηγαίο κώδικα και των δύο. Η Sleepycat προσφέρει υπηρεσίες συμβουλευτικής και ανάπτυξης για τους πελάτες της, αλλά τις προσφέρει υπό άδειες που τους επιτρέπουν να ενσωματώνονται χωρίς την υποχρέωση διανομής του Πηγαίου Κώδικα. Ασφαλώς, το πράττει μέσα στα πλαίσια ενός συγκεκριμένου συμβολαίου, και γενικότερα, μέσα σε ένα καθεστώς πωλήσεων ιδιοταγούς λογισμικού. Το 2005, η Sleepycat Software εξαγοράστηκε από την Oracle.

5.2.6. Πώληση εμπορικού σήματος

Αν και είναι εφικτό να αποκτήσει κανείς πολύ παρόμοια προϊόντα έναντι πολύ λιγότερων χρημάτων, πολλοί πελάτες προτίθενται να πληρώσουν επιπλέον για να αγοράσουν ένα εμπορικό σήμα. Αυτή η αρχή υιοθετείται από πολλές επιχειρήσεις, που επενδύουν στην καθιέρωση μίας μάρκας που διαθέτει μία καλή και πολύ αναγνωρίσιμη εικόνα, η οποία τους επιτρέπει, στη συνέχεια, να πωλούν ελεύθερα προϊόντα με ένα επαρκές περιθώριο κέρδους. Σε πολλές περιπτώσεις, δεν πωλούν απλώς αυτά τα προϊόντα, αλλά και υπηρεσίες, που γίνονται δεκτές από τους πελάτες ως μία προστιθέμενη αξία. Οι γνωστότερες περιπτώσεις αυτού του επιχειρηματικού μοντέλου είναι οι εταιρείες που πωλούν διανομές του GNU/Linux. Αυτές οι εταιρείες προσπαθούν να πωλήσουν κάτι που είναι δυνατόν να προμηθευθεί κανείς, σε πολύ χαμηλότερο κόστος, από το Διαδίκτυο (ή από άλλες πηγές με μία κατώτερη εταιρική εικόνα). Συνεπώς, πρέπει να οδηγήσουν τους καταναλωτές να αναγνωρίζουν το εμπορικό τους σήμα και να είναι διατεθειμένοι να πληρώσουν το πρόσθετο κόστος. Για να το πετύχουν, δεν επενδύουν απλώς στη διαφήμιση, αλλά προσφέρουν και αντικειμενικά πλεονεκτήματα (όπως, για παράδειγμα, μία καλοστημένη διανομή ή ένα κανάλι διανομής που βρίσκεται κοντά στον πελάτη). Πέραν αυτού, προσφέρουν συνήθως μία ευρεία ποικιλία υπηρεσιών γύρω από αυτό (από εκπαίδευση, μέχρι προγράμματα πιστοποίησης τρίτων), για να εκμεταλλευθούν όσο περισσότερο γίνεται το εταιρικό σήμα.

Παράδειγμα

Η Red Hat (<http://www.redhat.com>) [56]. Η Red Hat Linux άρχισε να διανέμεται το 1994 (η εταιρεία άρχισε να γίνεται γνωστή με το τρέχον όνομά της κατά το 1995). Για πολύ καιρό, η Red Hat κατάφερε να καθιερώσει το όνομά της ως η

κατ' εξοχήν διανομή GNU/Linux (αν και γύρω από τα μέσα του 2000, μοιράσθηκε αυτή τη θέση με άλλες εταιρείες, όπως την OpenSUSE, την Ubuntu, και, ίσως, την Debian). Αρκετά χρόνια αργότερα, η Red Hat πωλεί κάθε είδους υπηρεσίες που σχετίζονται με την διανομή, το GNU/Linux και το Ελεύθερο Λογισμικό γενικά.

5.3. Άλλες κατατάξεις επιχειρηματικών μοντέλων

Η βιβλιογραφία για το Ελεύθερο Λογισμικό προσφέρει άλλες κατατάξεις των παραδοσιακών επιχειρηματικών μοντέλων. Υπό μορφή παραδείγματος, παραθέτουμε εδώ μερικά από αυτά.

5.3.1. Ταξινόμηση κατά Hecker

Η κατάταξη που παρουσιάστηκε στο βιβλίο "Setting up shop: the business of open source software" (Frank Hecker, 1998) [141] χρησιμοποιήθηκε πολύ στην διαφήμιση της Open Source Initiative, και, επίσης, ήταν μία από τις πρώτες που προσπάθησαν να κατατάξουν τις επιχειρήσεις που προέκυψαν εκείνη την εποχή. Περιλαμβάνει, πάντως, διάφορα μοντέλα, που έχουν μικρή σχέση με το Ελεύθερο Λογισμικό (όπου το Ελεύθερο Λογισμικό είναι ελάχιστα περισσότερο από έναν απλό συνοδό του κυρίως μοντέλου). Σε κάθε περίπτωση, τα μοντέλα που περιγράφει έχουν ως εξής:

- *Πωλήσεις Υποστήριξης* (πώληση υπηρεσιών που σχετίζονται με το προϊόν). Η εταιρεία προωθεί ένα προϊόν ελεύθερου λογισμικού (το οποίο και ανέπτυξε, ή στο οποίο συμμετέχει ενεργώς) και πωλεί συμβουλευτικές υπηρεσίες, ή την προσαρμογή του λογισμικού σε ειδικές απαιτήσεις.
- *Ηγέτης Απώλειας ή "Κράχτης"* (*Loss leader*, πώληση άλλων ιδιοταγών προϊόντων). Σε αυτή την περίπτωση, το Ελεύθερο πρόγραμμα χρησιμοποιείται για μία, κατά κάποιο τρόπο, προώθηση της πώλησης άλλων ιδιοταγών προϊόντων που σχετίζονται με αυτό.
- *Πάγωμα Widget* (Πώληση υλικού *hardware*). Η βασική επιχειρηματική δραστηριότητα έγκειται στην πώληση *hardware*, ενώ το ελεύθερο λογισμικό θεωρείται ένα συμπλήρωμα που μπορεί να βοηθήσει την εταιρεία να κερδίσει ένα ανταγωνιστικό πλεονέκτημα.
- *Πώληση παράπλευρων προϊόντων* (*Accessorising*). Πωλούνται προϊόντα που σχετίζονται με το Ελεύθερο Λογισμικό, όπως π.χ. βιβλία, συσκευές για υπολογιστές, κλπ.
- *Ενεργοποιητής υπηρεσιών* (*Service enabler*, πώληση υπηρεσιών). Το Ελεύθερο Λογισμικό χρησιμεύει για τη δημιουργία μίας υπηρεσίας (που συνήθως είναι προσβάσιμη online, μέσω του Διαδικτύου) από την οποία η εταιρεία αντλεί κέρδος.
- *Παραχώρηση δικαιωμάτων εμπορικού σήματος* (*Brand licensing*, πώληση εμπορικού σήματος). Μία εταιρεία κατοχυρώνει εμπορικά σήματα (*trademarks*), τα οποία καταφέρει να συνδέσει με προγράμματα Ελεύθερου Λογισμικού, που η ίδια, ενδεχομένως, ανέπτυξε. Κατόπιν, προσπορίζεται έσοδα, μέσω της αδειοδότησης των εμπορικών αυτών σημάτων.
- *Πούλησέ το, ελευθέρωσέ το* (*sell it, free it*). Πρόκειται για ένα μοντέλο που είναι παρόμοιο με το μοντέλο του Ηγέτη Απώλειας (*Loss leader*), αλλά με εφαρμογή κατά κυκλικό τρόπο. Κατ' αρχήν, ένα προϊόν προωθείται ως Ελεύθερο Λογισμικό. Αν αποδειχθεί σχετικά επιτυχημένο, η επόμενη έκδοση διανέμεται, για ένα διάστημα, ως ιδιοταγές λογισμικό, μετά την λήξη του οποίου απελευθερώνεται. Σε εκείνη τη φάση, μία νέα ιδιοταγής έκδοση διανέμεται, και ούτω καθ' εξής, με επιτυχία.

- *Franchising Λογισμικού*. Μία εταιρεία παραχωρεί τη δικαιοχρησία χρήσης των εμπορικών της σημάτων, σχετικά με ένα ελεύθερο πρόγραμμα.

Σημείωση

Οι αναγνώστες θα έχουν προσέξει ότι αυτή η κατάταξη είναι αρκετά διαφορετική από εκείνη που εμείς παρουσιάσαμε, αλλά ακόμη και έτσι, μερικές κατηγορίες συμπίπτουν σχεδόν πλήρως με τις προβλεπόμενες από εμάς.

5.4. Επίπτωση σε μονοπωλιακές καταστάσεις

Η αγορά λογισμικού τείνει προς την κυριαρχία ενός προϊόντος σε κάθε τομέα. Οι χρήστες θέλουν να επωφεληθούν όσο γίνεται από την καταβαλλόμενη προσπάθεια εκμάθησης της λειτουργίας ενός προγράμματος, οι εταιρείες θέλουν να προσλάβουν ανθρώπους που να είναι εξοικειωμένοι με τη χρήση του λογισμικού τους και όλοι θέλουν τα δεδομένα τους να είναι διαχειρίσιμα από τα προγράμματα των εταιρειών και τους ανθρώπους με τους οποίους εργάζονται. Αυτός είναι ο λόγος για τον οποίο, η οποιαδήποτε πρωτοβουλία που έχει σχεδιασθεί για να διακόψει μία *de facto* κατάσταση, με ένα προϊόν να κυριαρχεί ξεκάθαρα στην αγορά, είναι προορισμένη να οδηγήσει σε μία αντίστοιχη κατάσταση: Αν είναι επιτυχής, τότε το νέο προϊόν θα έρθει να πάρει τη θέση του, και εντός μικρού χρονικού διαστήματος, θα έχουμε και πάλι ένα νέο κυρίαρχο προϊόν.

Μόνον οι τεχνολογικές αλλαγές οδηγούν, για μία μικρή περίοδο, σε μία επαρκή αστάθεια, ώστε να μην κυριαρχεί, ξεκάθαρα, κανείς. Το γεγονός, όμως, της ύπαρξης ενός κυρίαρχου προϊόντος, δε χρειάζεται να οδηγήσει απαραίτητως στη δημιουργία ενός επιχειρηματικού μονοπωλίου. Η βενζίνη, για παράδειγμα, είναι ένα προϊόν που σχεδόν κυριαρχεί στην αγορά καυσίμων για ιδιωτικά οχήματα, αλλά (σε μία ελεύθερη αγορά βενζίνης) υπάρχουν πολλές εταιρείες παραγωγής και διανομής για το ίδιο προϊόν. Στην πραγματικότητα, όταν συζητάμε για λογισμικό, το ανησυχητικό είναι όταν ένα προϊόν συμβαίνει να καταφέρνει να κυριαρχεί στην αγορά, διότι το προϊόν αυτό έχει μόνον έναν διαθέσιμο προμηθευτή. Το Ελεύθερο Λογισμικό προσφέρει μία εναλλακτική λύση σε αυτή την κατάσταση: τα ελεύθερα προϊόντα μπορούν να προωθούνται από μία συγκεκριμένη εταιρεία, αλλά αυτή η εταιρεία δεν τα ελέγχει ή, τουλάχιστον, όχι στο βαθμό που το ιδιοταγές λογισμικό μας συνήθισε. Στον κόσμο του Ελεύθερου Λογισμικού, ένα κυρίαρχο προϊόν δεν συνεπάγεται απαραίτητως το μονοπώλιο μίας εταιρείας. Αντιθέτως, ανεξαρτήτως από το ποιο προϊόν κυριαρχεί στην αγορά, πολλές εταιρείες μπορούν να ανταγωνισθούν στη διάθεσή του, στη βελτίωσή του, στην προσαρμογή του στις ανάγκες του πελάτη και στην προσφορά υπηρεσιών που σχετίζονται με αυτό.

5.4.1. Στοιχεία που ενοούν τα κυρίαρχα προϊόντα

Στο λογισμικό υπολογιστών, είναι συχνό να έχουμε ένα σαφώς κυρίαρχο προϊόν σε κάθε τομέα της αγοράς. Και αυτό είναι φυσιολογικό για διάφορους λόγους, μεταξύ των οποίων θα μπορούσαμε να τονίσουμε τα εξής:

- *Μορφές δεδομένων*. Σε πολλές περιπτώσεις, η μορφή δεδομένων (*format*) συσχετίζεται πολύ στενά με μία εφαρμογή. Όταν ένας επαρκώς μεγάλος αριθμός χρηστών τη χρησιμοποιήσει, τότε αυτή η μορφή δεδομένων αναδεικνύεται σε μία εκ των πραγμάτων πρότυπη μορφή και οι πιέσεις για την υιοθέτησή της (και, συνεπώς, και για την καθιέρωσή της) γίνονται τεράστιες.

- *Αλυσίδες διανομής*. Ένα από τα προβλήματα στην έναρξη της χρήσης ενός λογισμικού, είναι, συνήθως, το να αποκτήσει κανείς ένα αντίγραφό του. Και καθίσταται, συνήθως, δύσκολο να βρει κανείς προγράμματα που να μην είναι ηγέτες στο δικό τους τομέα αγοράς. Οι αλυσίδες διανομής

είναι δαπανηρές στη συντήρησή τους, υπό την έννοια ότι, για τους ανταγωνιστές της μειοψηφίας, καθίσταται δύσκολο να φθάσουν ως το ράφι των καταστημάτων υπολογιστών, όπου ο τελικός χρήστης να μπορεί να τα αγοράσει. Για το κυρίαρχο, όμως, προϊόν, αυτό είναι εύκολο: ο πρώτος ενδιαφερόμενος να το αποκτήσει, θα είναι το ίδιο το κατάστημα υπολογιστών.

- **Προώθηση (Marketing).** Άπαξ και ένα προϊόν αρχίσει να χρησιμοποιείται από σημαντική μερίδα πληθυσμού, τότε η "δωρεάν" προώθηση που κερδίζει, είναι τεράστια. Η "από στόμα σε στόμα" διάδοσή του λειτουργεί, επίσης, πολύ καλά, όταν ρωτάμε και ανταλλάσσουμε πληροφορίες με τους ανθρώπους που γνωρίζουμε. Αλλά προ πάντων, η επιρροή των MME είναι υπέρμετρη: τα περιοδικά υπολογιστών θα αναφερθούν ξανά και ξανά σε ένα προϊόν, εάν αυτό μοιάζει να το χρησιμοποιούν οι περισσότεροι. Θα δημιουργηθούν σειρές μαθημάτων εκπαίδευσης γύρω από αυτό, βιβλία που να το περιγράφουν, συνεντεύξεις με χρήστες, κλπ.

- **Επένδυση στην εκπαίδευση (training).** Όταν έχουμε ήδη επενδύσει χρόνο και χρήμα για την εκμάθηση λειτουργίας ενός εργαλείου, τότε προκύπτει ένα ισχυρό κίνητρο για τη μη αλλαγή αυτού του εργαλείου. Εξ' άλλου, αυτό το εργαλείο είναι, συνήθως, εκείνο που ήδη κυριαρχεί στην αγορά, διότι είναι ευκολότερο να βρει κανείς ανθρώπους και εκπαιδευτικά υλικά για τη διδασκαλία της χρήσης του.

- **Προεγκατεστημένο λογισμικό.** Η απόκτηση ενός μηχανήματος με προεγκατεστημένο λογισμικό αποτελεί, σίγουρα, ένα μέγα κίνητρο προς την χρήση του, ακόμη και αν πρέπει να πληρωθεί χωριστά. Και συνήθως, το είδος λογισμικού που ο πωλητής του μηχανήματος θα είναι προδιατεθειμένος να προεγκαταστήσει, θα είναι μόνο το περισσότερο χρησιμοποιούμενο.

5.4.2. Ο κόσμος του ιδιοταγούς λογισμικού

Στον κόσμο του ιδιοταγούς λογισμικού, η εμφάνιση ενός κυρίαρχου προϊόντος σε οποιονδήποτε τομέα της αγοράς, ισοδυναμεί με ένα μονοπώλιο εκ μέρους της εταιρείας που το παράγει. Έχουμε, για παράδειγμα, αυτές τις *de facto* (ή σχεδόν) μονοπωλιακές καταστάσεις ενός προϊόντος και μίας εταιρείας, στην αγορά των λειτουργικών συστημάτων, των προγραμμάτων desktop publishing, των βάσεων δεδομένων, του γραφιστικού σχεδίου, των επεξεργαστών κειμένου, των λογιστικών φύλλων, κλπ. Και αυτό ισχύει διότι η εν λόγω εταιρεία ασκεί τεράστιο έλεγχο πάνω στο ηγετικό προϊόν, σε τέτοιο βαθμό, που μόνον αυτοί μπορούν να κατευθύνουν την εξέλιξή του, τις βασικές κατευθυντήριες γραμμές πάνω στις οποίες θα αναπτυχθεί, την ποιότητά του, κλπ. Οι χρήστες διαθέτουν πολύ μικρό βαθμό ελέγχου, αφού έχουν ελάχιστο κίνητρο για να λάβουν υπ' όψη τους άλλα προϊόντα (για λόγους που μνημονεύσαμε στο προηγούμενο κεφάλαιο). Από αυτή την άποψη, ο ανταγωνισμός ελάχιστα πράγματα μπορεί να κάνει, εκτός από το να προσπαθεί να αμφισβητήσει την κυρίαρχη θέση του προϊόντος, μέσω της βελτίωσης των δικών του προϊόντων, (αντιρροπώντας τις ίδιες τις αιτίες), και συνήθως με περιορισμένη επιτυχία. Αυτή η κατάσταση θέτει όλον αυτόν τον τομέα αγοράς στα χέρια της στρατηγικής της κυρίαρχης εταιρείας. Όλοι οι παίκτες εξαρτώνται από αυτήν, και ακόμη και η ανάπτυξη τεχνολογίας λογισμικού σε αυτόν τον τομέα, θα προβληθεί αναλόγως των βελτιώσεων που επιφέρει στο δικό της προϊόν. Σε γενικές γραμμές, αυτή είναι μία κατάσταση όπου αναδεικνύονται οι χειρότερες οικονομικές επιπτώσεις ενός μονοπωλίου, και ειδικότερα, το μειωμένο κίνητρο για την κυρίαρχη εταιρεία να προσαρμόζει τα προϊόντα στις συνεχώς εξελισσόμενες ανάγκες των πελατών της, αφού μετετράπησαν σε μία εγκλωβισμένη αγορά (captive market).

5.4.3. Η κατάσταση με το Ελεύθερο Λογισμικό

Στην περίπτωση, όμως, του Ελεύθερου Λογισμικού, ή ύπαρξη ενός κυρίαρχου προϊόντος δεν μεταφράζεται αυτομάτως σε ένα επιχειρηματικό μονοπώλιο. Αφού το προϊόν είναι ελεύθερο, οποιαδήποτε εταιρεία μπορεί να το επεξεργασθεί, να το βελτιώσει, να το προσαρμόσει στις ανάγκες των πελατών της, και γενικότερα, να βοηθήσει στην ανάπτυξή του. Εξ' άλλου, ακριβώς χάρη στην κυρίαρχη θέση του, θα προκύψουν πολλές εταιρείες που θα ενδιαφέρονται να δουλέψουν πάνω σε αυτό. Αν ο “αρχικός” δημιουργός (δηλαδή η εταιρεία που πρωτοανέπτυξε το προϊόν) επιθυμεί να παραμείνει στην αγορά, τότε θα πρέπει να ανταγωνισθεί με όλες τις υπόλοιπες και, συνεπώς, θα έχει πολύ μεγάλο κίνητρο να φροντίσει να εξελιχθεί το προϊόν προς την κατεύθυνση ακριβώς που επιθυμούν οι χρήστες.

Θα διαθέτει, ασφαλώς, το πλεονέκτημα της καλύτερης γνώσης του προγράμματος, αλλά δεν τελειώνουν όλα εδώ. Θα πρέπει να ανταγωνισθούν για τον κάθε πελάτη. Συνεπώς, η εμφάνιση κυρίαρχων προϊόντων στον κόσμο του Ελεύθερου Λογισμικού, μεταφράζεται σε μεγαλύτερο ανταγωνισμό ανάμεσα στις εταιρείες. Και μέσα σε αυτό το περιβάλλον, οι χρήστες ανακτούν τον έλεγχο: εφ' όσον οι ανταγωνιζόμενες εταιρείες επιθυμούν να επιβιώσουν, δεν μπορούν να πράξουν τίποτε άλλο παρά να αφουγκρασθούν τους χρήστες. Και είναι ακριβώς αυτό που θα διασφαλίσει ότι το προϊόν θα βελτιώνεται.

Ελεύθερα προϊόντα που είναι κυρίαρχα στον τομέα τους

Για πολύ καιρό, η Apache ήταν ο ηγέτης στην αγορά των διακομιστών διαδικτύου. Αλλά υπάρχουν πολλές εταιρείες πίσω από την Apache, ξεκινώντας από κάποιες πολύ μεγάλες (σαν την IBM) και φθάνοντας σε άλλες πολύ μικρότερες. Και όλες τους δεν έχουν άλλη επιλογή παρά να ανταγωνίζονται μέσω της βελτίωσης και, συνήθως, μέσω της συνεισφοράς των βελτιώσεών τους προς το project. Παρά το γεγονός ότι η Apache αποτελεί σχεδόν ένα μονοπώλιο σε πολλούς τομείς (π.χ. είναι ο μόνος τύπος διακομιστή που λαμβάνεται υπ' όψη για πλατφόρμες GNU/Linux ή *BSD), δεν εξαρτάται από μία μόνον εταιρεία, αλλά, στην κυριολεξία, από καμία δεκαριά από αυτές.

Οι διάφορες διανομές GNU/Linux αποτελούν, επίσης, μία ενδιαφέρουσα περίπτωση. Το σύστημα GNU/Linux δεν είναι, ασφαλώς, ένα μονοπώλιο, αλλά αποτελεί την εν δυνάμει δεύτερη επιλογή στην αγορά των λειτουργικών συστημάτων. Και παρά ταύτα, η κατάσταση δεν οδηγήθηκε σε απόλυτο έλεγχο από μία μόνο εταιρεία. Αντιθέτως, υπάρχουν δεκάδες διανομών, που τις κατασκευάζουν διαφορετικές εταιρείες, οι οποίες και ανταγωνίζονται ελεύθερα στην αγορά. Κάθε μία από αυτές προσπαθεί να προσφέρει βελτιώσεις, τις οποίες οι ανταγωνιστές της θα πρέπει να υιοθετήσουν, διαφορετικά θα διατρέξουν τον κίνδυνο να μείνουν εκτός αγοράς. Επιπλέον, δεν μπορούν να απομακρυνθούν και πάρα πολύ από τις “κλασικές προδιαγραφές” του GNU/Linux, καθότι οι χρήστες θα απέρριπταν κάτι τέτοιο, θεωρώντας το ως μία “απόκλιση από το κανονικό”. Ο τρόπος με τον οποίο έχει διαμορφωθεί η κατάσταση, έπειτα από πολλά χρόνια αυξανόμενου μεριδίου αγοράς για το GNU/Linux, μας δείχνει ότι υπάρχουν δεκάδες εταιρειών που ανταγωνίζονται και, έτσι, επιτρέπουν στο σύστημα να εξελίσσεται. Για μία ακόμη φορά, όλες τους αγωνίζονται να ικανοποιήσουν τις απαιτήσεις των χρηστών. Αυτός είναι ο μόνος τρόπος που μπορούν να παραμείνουν στο παιχνίδι της αγοράς.

Ο GCC είναι ένα κυρίαρχο προϊόν στον κόσμο των μεταγλωττιστών C και C++ για την αγορά GNU/Linux. Και όμως, αυτό δεν οδήγησε, ακόμη, σε καμία μονοπωλιακή κατάσταση, παρά το ότι η Cygnus (τώρα γνωστή ως Red Hat) ήταν για πολύ καιρό υπεύθυνη για το συντονισμό της ανάπτυξής του. Υπάρχουν πολλές εταιρείες που επιφέρουν βελτιώσεις στο σύστημα και όλες τους ανταγωνίζονται, η κάθε μία στο δικό της κομμάτι αγοράς, για να ικανοποιήσουν τις επιθυμίες των δικών τους χρηστών. Στην πραγματικότητα, όταν κάποια εταιρεία ή οργανισμός αποτυγχάνει στον συντονισμό (ή ορισμένοι χρήστες το αντιλαμβάνονται κατ' αυτόν τον τρόπο), τότε δημιουργούνται οι προϋποθέσεις δημιουργίας μίας διακλάδωσης (fork) του project, που θα περιλαμβάνει, πλέον, δύο προϊόντα συγχρόνως, για ένα μικρό διάστημα, μέχρι να επανενωθούν και πάλι (όπως συμβαίνει τώρα με το GCC 3.x).

5.4.4. Στρατηγικές για δημιουργία μονοπωλίων με το Ελεύθερο Λογισμικό

Παρά το ότι ο κόσμος του Ελεύθερου Λογισμικού είναι πολύ πιο εχθρικός στη δημιουργία

επιχειρηματικών μονοπωλίων από ότι είναι ο κόσμος του ιδιοταγούς λογισμικού, υπάρχουν στρατηγικές που μπορεί να χρησιμοποιήσει μία εταιρεία, για να προσπαθήσει να προσεγγίσει μία κατάσταση μονοπωλιακής κυριαρχίας μίας αγοράς. Αυτές οι πρακτικές είναι πολύ συνηθισμένες σε πολλούς τομείς της οικονομικής ζωής και, για να μπορέσουμε να τις προλάβουμε, δημιουργήσαμε εποπτικά σώματα που ρυθμίζουν τον ανταγωνισμό, εξ' ου και δεν θα υπεισέλθουμε σε πολλές λεπτομέρειες σχετικά με αυτά. Εν τούτοις, θα μνημονεύσουμε έναν συγκεκριμένο τρόπο, που μέχρι ενός βαθμού, είναι χαρακτηριστικός για την αγορά του λογισμικού και ο οποίος έχει, ήδη, δοκιμασθεί σε κάποιες καταστάσεις: η αποδοχή πιστοποιήσεων προϊόντων από τρίτους.

Όταν μία εταιρεία επιθυμεί να διανείμει ένα προϊόν λογισμικού (ελεύθερο ή ιδιοταγές) που λειτουργεί σε συνδυασμό με άλλα, είναι συνηθισμένο να “πιστοποιεί” αυτό το προϊόν για ένα συγκεκριμένο συνδυασμό. Ο κατασκευαστής αναλαμβάνει να προσφέρει υπηρεσίες (αναβαθμίσεις, υποστήριξη, επίλυση προβλημάτων, κλπ.) μόνο εάν ο πελάτης εγγυηθεί ότι το προϊόν Ελεύθερου Λογισμικού χρησιμοποιείται στα πλαίσια ενός πιστοποιημένου περιβάλλοντος.

Για παράδειγμα, ο κατασκευαστής ενός προγράμματος διαχείρισης μίας βάσης δεδομένων μπορεί να πιστοποιήσει το προϊόν του για μία ορισμένη διανομή GNU/Linux, και για καμία άλλη. Αυτό συνεπάγεται ότι οι πελάτες του θα πρέπει να χρησιμοποιήσουν εκείνη τη συγκεκριμένη διανομή GNU/Linux, ειδικά, μπορούν να ξεχάσουν την υποστήριξη του κατασκευαστή (η οποία, όταν πρόκειται για ένα ιδιοταγές προϊόν, μπορεί να καταστεί πρακτικώς ανέφικτη). Αν ένας συγκεκριμένος κατασκευαστής καταφέρει να πετύχει μία αμιγώς κυρίαρχη θέση για ένα προϊόν πιστοποιημένο από τρίτους, τότε οι χρήστες δεν θα έχουν καμία άλλη επιλογή από τη χρήση αυτού του προϊόντος. Εάν σε αυτόν τον τομέα αγοράς είναι σημαντική η πιστοποίηση, τότε θα βρεθούμε, ξανά, μπροστά σε μία κατάσταση επιχειρηματικού μονοπωλίου.

Σημείωση

Μέχρι ενός σημείου, εντός της αγοράς των διανομών GNU/Linux, αρχίζουμε να βλέπουμε λίγες περιπτώσεις καταστάσεων που τείνουν προς ένα *de facto* μονοπώλιο, μέσω της πιστοποίησης. Για παράδειγμα, υπάρχουν πολλοί κατασκευαστές ιδιοταγών προϊόντων που τα πιστοποιούν μόνο για μία συγκεκριμένη διανομή GNU/Linux (που πολύ συχνά είναι η Red Hat Linux). Προς το παρόν, αυτό δεν φαίνεται να οδηγεί σε μία κατάσταση μονοπωλίου για καμία εταιρεία, πράγμα που μπορεί να οφείλεται στο γεγονός ότι η πιστοποίηση δεν είναι και τόσο σημαντική για τους χρήστες που βρίσκονται στην αγορά των διανομών GNU/Linux. Αλλά μόνο το μέλλον θα δείξει εάν αυτή η κατάσταση θα πλησιάζει, σε κάποιο σημείο, ένα *de facto* μονοπώλιο. \

Εν τούτοις, είναι σημαντικό να θυμόμαστε δύο σχόλια σχετικά με τα παραπάνω. Το πρώτο είναι ότι δεν θα είναι καθόλου εύκολο να κατακτηθούν αυτές οι θέσεις μονοπωλίου και, σε κάθε περίπτωση, θα επιτευχθούν μέσω μηχανισμών, γενικά “μη-συσχετιζόμενων με το λογισμικό” (αντιθέτως με εκείνο που ισχύει σε μία κατάσταση με ένα κυρίαρχο προϊόν, που από όσα είδαμε, είναι σχετικά συνηθισμένη και επιτυγχάνεται με μηχανισμούς που σχετίζονται άμεσα με την Πληροφορική/IT και τα μοντέλα χρήσης της). Το δεύτερο είναι ότι, αν όλα τα λογισμικά που χρησιμοποιούνται είναι Ελεύθερα, τότε αυτή στρατηγική έχει ελάχιστες πιθανότητες επιτυχίας, αν όχι μηδαμινές. Ένας κατασκευαστής μπορεί να καταφέρει να πιστοποιήσει τα προϊόντα του από πολλές εταιρείες, αλλά, αν οι πελάτες το θεωρούν σκόπιμο, θα έχουν πάντα τη δυνατότητα να αναζητήσουν διαφορετικές εταιρείες, που να προσφέρουν διαφορετικές υπηρεσίες και υποστήριξη, σε σύγκριση με εκείνες που έκαναν την πιστοποίηση.

6. Το Ελεύθερο Λογισμικό και οι Δημόσιοι Οργανισμοί

"[...] Για να μπορεί ένα λογισμικό να είναι αποδεκτό για το Κράτος, δε χρειάζεται μόνο να είναι τεχνικώς ικανό να επιτελεί ένα έργο, αλλά πρέπει και οι όροι προκήρυξης για την απόκτησή του να πληρούν μία σειρά απαιτήσεων σχετικά με την αδειοδότηση, χωρίς τις οποίες το Κράτος δεν μπορεί να εγγυηθεί προς τους πολίτες του ότι τα δεδομένα τους υφίστανται κατάλληλη επεξεργασία, με τον απαραίτητο σεβασμό προς την εμπιστευτικότητα και τη διαχρονική προσβασιμότητα, καθότι αυτές είναι ιδιαιτέρως κρίσιμες πλευρές των φυσιολογικώς αναμενομένων υποχρεώσεών του." Edgar David Villanueva Núñez (απαντητική επιστολή προς τον Γεν. Διευθυντή της Microsoft Peru, 2001)

Οι Δημόσιοι θεσμικοί οργανισμοί, τόσο εκείνοι με τη Νομοθετική εξουσία, όσο και εκείνοι που είναι υπεύθυνοι για την διαχείριση του Κράτους (οι "Δημόσιοι Διαχειριστές"), παίζουν έναν πολύ σημαντικό ρόλο, σε ότι αφορά την υιοθεσία και την προώθηση τεχνολογιών. Αν και μέχρι το έτος 2000 αυτοί οι θεσμοί δεν επέδειξαν, πρακτικώς, κανένα ενδιαφέρον για το φαινόμενο του Ελεύθερου Λογισμικού (πλην μερικών εξαιρέσεων), έκτοτε η κατάσταση άρχισε να αλλάζει. Από τη μία πλευρά, πολλοί Δημόσιοι Οργανισμοί άρχισαν να χρησιμοποιούν το Ελεύθερο Λογισμικό ως τμήμα της υποδομής IT που διαθέτουν. Από την άλλη πλευρά, μέσα από τον ρόλο τους ως προωθητών της Κοινωνίας της Πληροφορίας, ορισμένοι άρχισαν να προωθούν, με τρόπο άμεσο ή έμμεσο, την ανάπτυξη και τη χρήση του Ελεύθερου Λογισμικού. Επιπλέον, μερικά Νομοθετικά σώματα άρχισαν να εστιάζουν την προσοχή τους (σιγά-σιγά) στο Ελεύθερο Λογισμικό, ενίοτε ενθαρρύνοντας την ανάπτυξή του, κάποιες άλλες φορές εμποδίζοντάς την, και ενίοτε, απλώς συνειδητοποιώντας/ αναγνωρίζοντας την ύπαρξή του.

Πριν εμβαθύνουμε στις λεπτομέρειες, είναι σημαντικό να θυμόμαστε ότι για πολύ καιρό, το Ελεύθερο Λογισμικό αναπτυσσόταν χωρίς επίσημη υποστήριξη (ή και χωρίς κανένα ενδιαφέρον) εκ μέρους των Δημόσιων θεσμικών οργανισμών. Για το λόγο αυτό, η προσοχή που προσελκύει από πολλούς εξ αυτών, δεν είναι χωρίς αμφισβήτηση και προβλήματα. Επίσης, κατά τα τελευταία λίγα χρόνια, κερδίζουν έδαφος οι πρωτοβουλίες που σχετίζονται με τα Ανοικτά Πρότυπα, που συχνά καταλήγουν σε μέτρα (κατά το μάλλον ή ήττον, με άμεσο τρόπο) συνδεδεμένα με το Ελεύθερο Λογισμικό.

Σε αυτό το κεφάλαιο, θα προσπαθήσουμε να περιγράψουμε την τρέχουσα κατάσταση και τις ιδιαιτερότητες του Ελεύθερου Λογισμικού, στο πλαίσιο της "δημόσιας" σφαίρας.

6.1. Επίπτωση στους Δημόσιους Οργανισμούς

Έχουν γίνει πολλές μελέτες που εστίαζαν στη χρήση του Ελεύθερου Λογισμικού, μέσα στα πλαίσια του Δημόσιου Τομέα (για παράδειγμα, "Open source software for the public administration" ("Λογισμικό Ανοικτού Κώδικα για την Δημόσια Διοίκηση"), 2004 [159]; "Open source software in e-Government, analysis and recommendations drawn up by a working group under the Danish board of technology" ("Λογισμικό Ανοικτού Κώδικα στην ηλεκτρονική διακυβέρνηση: Ανάλυση και συστάσεις που προέκυψαν από την Ομάδα Εργασίας του Δανέζικου Οργανισμού Τεχνολογίας"), 2002 [180]; "Free Software/ Open Source: information society opportunities for Europe?" ("ΕΛ/ΛΑΚ: Ευκαιρίες για την Κοινωνία της Πληροφορίας στην Ευρώπη;"), 1999 [132], και "The case for government promotion of Open Source software" ("Η υπόθεση της κυβερνητικής προώθησης του Λογισμικού Ανοικτού Κώδικα") , 1999 [213]). Στο επόμενο κεφάλαιο, θα συζητήσουμε μία από τις πιο αξιοσημείωτες (τόσο με τη θετική, όσο και με την αρνητική έννοια).

6.1.1. Πλεονεκτήματα και παράπλευρα θετικά οφέλη

Μερικά από τα οφέλη της χρήσης Ελεύθερου Λογισμικού από τους Δημόσιους Οργανισμούς και οι κυριότερες νέες προοπτικές που ανοίγονται, έχουν ως εξής:

1) Ανάπτυξη τοπικής βιομηχανίας

Ένα από τα μεγαλύτερα οφέλη του Ελεύθερου Λογισμικού είναι η δυνατότητα ανάπτυξης μίας εντόπιας βιομηχανίας λογισμικού. Όταν χρησιμοποιούμε ιδιοταγές λογισμικό, όλα τα χρήματα που δαπανούμε για Άδειες χρήσης, πάνε κατ' ευθείαν στον κατασκευαστή του προϊόντος, και κάθε αγορά ενισχύει την θέση αυτού του κατασκευαστή, πράγμα που δεν είναι κατ' ανάγκην αρνητικό, αλλά δεν είναι πολύ αποτελεσματικό για την περιοχή με την οποία η δημόσια διοίκηση συνδέεται, όταν το συγκρίνουμε με την εναλλακτική λύση της χρήσης ενός ελεύθερου προγράμματος.

Σε αυτή την περίπτωση, οι εντόπιες εταιρείες θα μπορούν να ανταγωνιστούν στην προσφορά υπηρεσιών (και του ίδιου του προγράμματος) προς τη Δημόσια Διοίκηση, υπό πολύ παρόμοιες συνθήκες με εκείνες που ισχύουν για οποιαδήποτε άλλη εταιρεία. Μπορούμε να πούμε ότι, κατά κάποιον τρόπο, η Δημόσια Διοίκηση “προλειαίνει το έδαφος” και καθιστά ευκολότερο για τον καθένα να ανταγωνίζεται μέσα σε αυτό το πλαίσιο. Ασφαλώς, ο όρος “καθένας” περιλαμβάνει τις εντόπιες εταιρείες, που θα έχουν την ευκαιρία να εκμεταλλευθούν τα ανταγωνιστικά τους πλεονεκτήματα (καλύτερη γνώση των αναγκών των πελατών, γεωγραφική εγγύτητα, κλπ.).

2) Ανεξαρτησία από έναν προμηθευτή και ανταγωνισμός αγοράς

Προφανώς, οποιαδήποτε οργάνωση θα προτιμήσει να εξαρτάται από μία ανταγωνιστική αγορά, παρά από ένα μοναδικό προμηθευτή, με την ικανότητα επιβολής των όρων υπό τους οποίους παρέχει το προϊόν του. Παρ' όλα αυτά, στον κόσμο της Δημόσιας Διοίκησης, αυτή η προτίμηση μετατρέπεται σε βασική προϋπόθεση, ή ακόμη και σε νομική υποχρέωση, μερικές φορές. Σε γενικές γραμμές, η Δημόσια Διοίκηση δεν μπορεί να επιλέξει να συμβληθεί με ένα δεδομένο προμηθευτή, παρά μάλλον, πρέπει να διευκρινίσει τις απαιτήσεις της με τέτοιο τρόπο, ώστε, εάν οποιαδήποτε ενδιαφερόμενη εταιρεία πληροί ορισμένα χαρακτηριστικά και προσφέρει το ζητούμενο προϊόν ή τη ζητούμενη υπηρεσία, να μπορεί να υποβάλει αίτηση για σύμβαση.

Για μια ακόμη φορά, στην περίπτωση του ιδιοταγούς λογισμικού, κάθε προϊόν διαθέτει μόνον ένα προμηθευτή (ακόμη και αν χρησιμοποιεί αρκετούς ενδιάμεσους). Αν δοθούν προδιαγραφές για ένα συγκεκριμένο προϊόν, τότε η Δημόσια Διοίκηση θα μπορεί και να αποφασίσει σε ποιον προμηθευτή να αναθέσει την σύμβαση. Σε πολλές περιπτώσεις, είναι στην κυριολεξία αδύνατη η αποφυγή διευκρίνισης προδιαγραφών για ένα δεδομένο προϊόν, όταν έχουμε να κάνουμε με προγράμματα υπολογιστών.

Για λόγους συμβατότητας στον ίδιο τον Δημόσιο Οργανισμό, ή την εξοικονόμηση στην εκπαίδευση και τη διαχείριση συστημάτων, ή και πολλά άλλα ακόμη, καθίσταται συχνό για ένα Δημόσιο Οργανισμό να αποφασίζει να χρησιμοποιήσει ένα συγκεκριμένο προϊόν. Ο μόνος τρόπος για να βγούμε από αυτή την κατάσταση, είναι να κατασκευασθεί το προϊόν με τις περιγραφείσες προδιαγραφές και να αποδοθεί ως ελεύθερο. Κατ' αυτό τον τρόπο, οποιαδήποτε ενδιαφερόμενη εταιρεία θα είναι σε θέση να προμηθεύει τόσο το ίδιο το προϊόν, όσο και οποιαδήποτε υπηρεσία σχετική με αυτό (με μόνο περιορισμό τις ικανότητες της εταιρείας και τη γνώση του προϊόντος). Εξ' άλλου, σε αυτή την περίπτωση σύμβασης, η Δημόσια Διοίκηση μπορεί να αλλάξει προμηθευτή στο μέλλον, εάν το επιθυμεί, ακόμη και χωρίς να υπάρχει κάποιο τεχνικό πρόβλημα, αφού ακόμη και όταν θα αλλάξει εταιρεία, θα συνεχίσει να χρησιμοποιεί το ίδιο προϊόν.

3) Ευελιξία και προσαρμοστικότητα σε ιδιαίτερες απαιτήσεις

Αν και κάθε εταιρεία που χρησιμοποιεί υπολογιστές χρειάζεται τη δυνατότητα προσαρμογής σε ειδικές ανάγκες, οι ιδιαιτερότητες της Δημόσιας Διοίκησης καθιστούν αυτό τον παράγοντα πολύ σημαντικό για την επιτυχή ενσωμάτωση ενός συστήματος λογισμικού. Στην περίπτωση του Ελεύθερου Λογισμικού, η προσαρμογή είναι ακόμη πιο εύκολη, και το σημαντικότερο, μπορεί να βασισθεί σε μία ανταγωνιστική αγορά, όπου απαιτείται η σύναψη συμβάσεων.

Όταν η Δημόσια Διοίκηση αγοράζει ένα ιδιοταγές προϊόν, η τροποποίησή του συνήθως συνεπάγεται τη σύναψη μίας συμφωνίας με τον κατασκευαστή, που είναι και ο μόνος που μπορεί νομίμως (και συχνά, και από τεχνικής πλευράς) να το πράξει. Υπό αυτές τις συνθήκες, είναι δύσκολο να διαπραγματευθεί κανείς, ιδιαίτερα εάν ο κατασκευαστής δεν τρέφει και τόσο μεγάλο ενδιαφέρον για την αγορά που του προσφέρει η συγκεκριμένη Δημόσια Διοίκηση. Αντιθέτως, με την χρήση ενός ελεύθερου προϊόντος, η Διοίκηση μπορεί να το τροποποιήσει όπως θέλει, υπό την προϋπόθεση ότι θα μισθώσει το κατάλληλο προσωπικό, ή θα αναθέσει εξωτερικά την τροποποίηση. Επί της αρχής, αυτή η εξωτερική ανάθεση μπορεί να γίνει προς οποιαδήποτε εταιρεία με τις ικανότητες και τη γνώση να το αναλάβει, που σημαίνει ότι είναι αναμενόμενο να ενδιαφερθούν διάφορες εταιρείες για τη διεκδίκηση. Φυσικά, αυτό οδηγεί σε χαμηλότερα κόστη και σε βελτίωση της ποιότητας.

Η περίπτωση των διανομών GNU/Linux

Κατά τα τελευταία λίγα χρόνια, στην Ισπανία, παρατηρήθηκε συχνά ορισμένες Διοικήσεις Περιφερειών να προχωρούν στη δημιουργία των δικών τους διανομών GNU/Linux. Αυτή η τάση άρχισε με το GNU/Linux, αλλά στις μέρες μας, υπάρχουν πολλές περισσότερες. Αν και μερικοί ειδήμονες άσκησαν κριτική σχετικά με την ύπαρξη αυτών των διανομών, αυτό αποτελεί σαφές παράδειγμα της προσαρμοστικότητας που επιτρέπει το Ελεύθερο Λογισμικό. Οποιοσδήποτε Δημόσιος οργανισμός, δαπανώντας σχετικά περιορισμένα κονδύλια, μπορεί να προχωρήσει σε μία τροποποίηση του GNU/Linux, προσαρμόζοντάς το στις ανάγκες του και τις προτιμήσεις του, χωρίς ουσιαστικώς κανένα όριο. Μπορεί, για παράδειγμα, να αλλάξει την εμφάνιση της επιφάνειας εργασίας, να επιλέξει ποιες θα είναι οι προεπιλεγμένες εφαρμογές και η γλώσσα του συστήματος, να βελτιώσει τη μετάφραση των διαφόρων εφαρμογών, κλπ. Με άλλα λόγια, εάν κριθεί επιθυμητό, τόσο η επιφάνεια εργασίας, όσο και οποιαδήποτε εφαρμογή λογισμικού που εκτελείται στον υπολογιστή, μπορούν να προσαρμοσθούν για να καλύψουν συγκεκριμένες απαιτήσεις.

Αυτή η προσαρμογή, θα επιφέρει, ασφαλώς, κάποια έξοδα, αλλά η εμπειρία απέδειξε ότι μπορεί να επιτευχθεί σχετικά οικονομικά, και η τάση μοιάζει να δείχνει ότι θα γίνεται όλο και πιο εύκολο (αλλά και πιο φθηνό) να δημιουργούνται προσαρμοσμένες διανομές.

4) Ευκολότερη υιοθέτηση των Ανοικτών Προτύπων

Με δεδομένη τη φύση τους, τα ελεύθερα προγράμματα χρησιμοποιούν συχνά ανοικτά ή μη-ιδιοταγή πρότυπα. Πράγματι, σχεδόν εξ' ορισμού, οποιαδήποτε πλευρά ενός Ελεύθερου προγράμματος και αν επιλέξουμε να δούμε, μπορεί να αναπαραχθεί πολύ εύκολα και, συνεπώς, δεν είναι ιδιοταγής. Για παράδειγμα, τα πρωτόκολλα που χρησιμοποιεί ένα ελεύθερο πρόγραμμα, για να επικοινωνεί με άλλα προγράμματα, μπορούν να μελετηθούν και να αναπαραχθούν και, υπό την έννοια αυτή, δεν είναι ούτε αυτά ιδιοταγή. Πέραν τούτου, πάρα πολύ συχνά και προς το συμφέρον των ίδιων των projects, προσπαθούμε να χρησιμοποιούμε Ανοικτά Πρότυπα.

Πάντως, ανεξαρτήτως των κινήτρων, είναι γεγονός ότι ότι τα ελεύθερα προγράμματα συνήθως χρησιμοποιούν μη-ιδιοταγή πρότυπα για την ανταλλαγή δεδομένων. Τα πλεονεκτήματα αυτού του γεγονότος για τη Δημόσια Διοίκηση είναι πολύ πιο εκτεταμένα από ότι για οποιονδήποτε άλλον οργανισμό, αφού η προώθηση ιδιοταγών προτύπων (έστω και έμμεσα, με τη χρήση τους) αποτελεί λόγο μεγάλης ανησυχίας στην περίπτωσή τους. Τέλος, από τουλάχιστον μία πλευρά, εκεί όπου υπεισέρχεται αλληλεπίδραση με τους πολίτες, η χρήση μη-ιδιοταγών προτύπων είναι βασική, αφού

δεν θα πρέπει να εξαναγκασθούν να αγοράσουν κανένα προϊόν από μία συγκεκριμένη εταιρεία, για να καταστεί εφικτή η αλληλεπίδρασή τους με τη Δημόσια Διοίκηση.

5) Δημόσιος έλεγχος των θεμάτων ασφαλείας

Για μία Δημόσια Διοίκηση, το να μπορεί να είναι σε θέση να εγγυηθεί ότι οι υπολογιστές της κάνουν μόνον εκείνο που πρέπει, συνιστά μία βασική υποχρέωση, και σε πολλές χώρες, αποτελεί επίσης και μία νομική προϋπόθεση. Συχνά, αυτά τα συστήματα χειρίζονται ιδιωτικά δεδομένα, για τα οποία κάποιοι τρίτοι θα μπορούσαν να ενδιαφέρονται (π.χ. φορολογικά στοιχεία, ποινικά μητρώα, αναγραφικά στοιχεία ή δεδομένα εθνικών απογραφών, εκλογικά δεδομένα, κλπ). Εάν γίνεται χρήση μία ιδιοταγούς εφαρμογής, χωρίς να είναι διαθέσιμος ο πηγαίος κώδικας, τότε είναι δύσκολο να παρασχεθούν εγγυήσεις ότι η εφαρμογή θα χειρισθεί τα ευαίσθητα δεδομένα με τον προσήκοντα τρόπο. Αλλά, ακόμη κι' αν μας παρασχεθεί ο πηγαίος κώδικας, οι πιθανότητες να μπορεί μία Δημόσια Διοίκηση να διασφαλίσει ότι δεν εμπεριέχεται *περίεργος* κώδικας, θα είναι πολύ περιορισμένες. Μόνον εάν το έργο που πρέπει να εκτελεσθεί μπορεί, συχνά και συστηματικά, να ανατίθεται σε τρίτους, και επιπλέον οποιοσδήποτε ενδιαφερόμενος μπορεί να το εξετάσει ενδελεχώς, τότε μόνον η Δημόσια Διοίκηση θα μπορεί να είναι ευλόγως σίγουρη ότι συμμορφώνεται με το καταστατικό της καθήκον, ή ότι τουλάχιστον λαμβάνει όλα τα απαραίτητα μέτρα που μπορεί, για να το πετύχει.

6) Μακροπρόθεσμη Διαθεσιμότητα

Πολλά από τα δεδομένα που οι Δημόσιες Υπηρεσίες επεξεργάζονται, καθώς και τα προγράμματα που χρησιμοποιούν για τους υπολογιστικούς αυτούς χειρισμούς, πρέπει να είναι διαθέσιμα επί ολόκληρες δεκαετίες. Είναι πάρα πολύ δύσκολο να εγγυηθεί κανείς ότι ένα ιδιοταγές πρόγραμμα θα συνεχίσει να είναι διαθέσιμο μετά από τόσο μεγάλο χρονικό διάστημα, ειδικά αν οι προσδοκίες μας είναι ότι αυτό θα πρέπει να συνεργάζεται με την εκάστοτε ισχύουσα πλατφόρμα, της τότε μελλοντικής εποχής. Αντιθέτως, είναι πιθανόν ο κατασκευαστής να έχει χάσει το ενδιαφέρον του για το προϊόν αυτό, χωρίς να το έχει προσαρμόσει στις νέες πλατφόρμες, ή μπορεί να δέχεται να το κάνει μόνον έναντι πολλών χρημάτων. Θα πρέπει και πάλι να θυμόμαστε ότι μόνον ο κατασκευαστής μπορεί να προσαρμόσει το προϊόν, άρα λοιπόν, οι διαπραγματεύσεις θα είναι δύσκολες. Στην περίπτωση του Ελεύθερου Λογισμικού, πάντως, η εφαρμογή είναι μετά βεβαιότητας διαθέσιμη, έτσι ώστε ο καθένας μπορεί να την προσαρμόσει και να την κάνει να λειτουργεί σύμφωνα με τις ανάγκες της εκάστοτε Δημόσιας υπηρεσίας. Αν αυτό δεν συμβεί αυθορμήτως, τότε η Δημόσια Διοίκηση διατηρεί πάντοτε το δικαίωμα να προκαλέσει το ενδιαφέρον πολλών εταιρειών, για να υποβάλλουν την συμφερότερη προσφορά για την εκτέλεση του έργου. Αυτό εγγυάται ότι τόσο η εφαρμογή όσο και τα δεδομένα που επεξεργάζεται θα παραμένουν διαθέσιμα όποτε απαιτείται.

7) Γενικότερη επίπτωση, πέραν της χρήσης από τη Δημόσια Διοίκηση

Πολλές εφαρμογές που χρησιμοποιούνται ή προωθούνται από διάφορες Δημόσιες Υπηρεσίες, χρησιμοποιούνται και σε πολλούς άλλους τομείς της κοινωνίας. Γι' αυτό το λόγο, κάθε δημόσια επένδυση για την ανάπτυξη ενός Ελεύθερου προϊόντος δεν ωφελεί μόνον την Δημόσια Αρχή αυτή καθ' εαυτή, αλλά ωφελεί και όλους τους πολίτες, που θα είναι σε θέση να χρησιμοποιούν το προϊόν για τις υπολογιστικές τους εργασίες, ενδεχομένως και με τις βελτιώσεις που επέφερε η ίδια η Δημόσια Αρχή.

Βιβλιογραφία

Οι αναγνώστες που ενδιαφέρονται για μία μελέτη πάνω στα πλεονεκτήματα του Ελεύθερου Λογισμικού για τη Δημόσια Διοίκηση, γραμμένο για τα Αμερικανικά δεδομένα του 1999, μπορούν να ανατρέξουν στο "The case for government promotion of open source software" (Mitch Stoltz, 1999) [213].

Σημείωση

Μία πολύ ιδιαίτερη περίπτωση, αλλά με τεράστια επίπτωση, που αναδεικνύει αυτή τη βέλτιστη χρήση των Δημόσιων πόρων, είναι η προσαρμογή των προγραμμάτων (program localisation) στις συνήθειες, την γλώσσα και το πολιτισμικό υπόβαθρο της κάθε τοπικής κοινότητας. Αν και η πιο εμφανής πλευρά των προγραμμάτων προσαρμογής είναι, σίγουρα, η μετάφραση του λογισμικού και του σχετικού εγχειριδίου χρήσης, υπάρχουν ωστόσο και άλλα θέματα που επηρεάζονται από αυτό (π.χ. από τη χρήση του συμβόλου για το τοπικό νόμισμα, περνώντας στην μορφή υπό την οποία εμφανίζονται η ημερομηνία και η ώρα για την συγκεκριμένη κοινότητα, μέχρι και τη χρήση παραδειγμάτων μέσα στα εγχειρίδια χρήσης, καθώς και τους ιδιωματικούς τρόπους έκφρασης που μπορούν να είναι κατανοητοί και αποδεκτοί από την κοινότητα αυτή).

Πάντως, αν μία Δημόσια Αρχή χρηματοδοτεί την προσαρμογή μίας συγκεκριμένης εφαρμογής, παραμετροποιώντας την στις δικές της ανάγκες, τότε είναι πολύ πιθανόν οι ανάγκες αυτές να συμπίπτουν ακριβώς με εκείνες των πολιτών της, υπό την έννοια ότι δε θα δημιουργήσει απλώς ένα πρόγραμμα που να καλύπτει τις δικές της ιδιαίτερες απαιτήσεις, αλλά θα καταστήσει, επίσης, διαθέσιμο κάτι που μπορεί να το έχει ο κάθε πολίτης, που ίσως το χρειάζεται, χωρίς κανένα πρόσθετο κόστος. Όταν, για παράδειγμα, μία Δημόσια Διοίκηση χρηματοδοτεί την προσαρμογή ενός λογισμικού για υπολογιστές σε μία γλώσσα που χρησιμοποιείται στη δική της εντόπια κοινότητα, δεν θα το χρησιμοποιεί μόνον η ίδια στα γραφεία της, αλλά θα είναι σε θέση και να το προσφέρει στους πολίτες της, με όλα όσα αυτό σημαίνει για την ανάπτυξη της Κοινωνίας της Πληροφορίας.

6.1.2. Δυσκολίες υιοθέτησης και άλλα προβλήματα

Από την άλλη μεριά, παρά το ότι υπάρχουν πολλά πλεονεκτήματα για την Δημόσια Διοίκηση από τη χρήση του Ελεύθερου Λογισμικού, υπάρχουν όμως και πολλές δυσκολίες που πρέπει να αντιμετωπισθούν όταν μία τέτοια μετάβαση πρέπει να εφαρμοσθεί. Από όλες αυτές τις διάφορες δυσκολίες, θα θέλαμε να αναφέρουμε τις εξής:

1) Έλλειψη γνώσης και Πολιτικής βούλησης

Το πρώτο πρόβλημα που το Ελεύθερο Λογισμικό συναντά, για να μπορέσει να εισχωρήσει στη Δημόσια Διοίκηση είναι κάτι που μοιράζεται, αναμφίβολα, από κοινού και με άλλες οργανώσεις: Το Ελεύθερο Λογισμικό παραμένει ακόμη ένας άγνωστος στους ανθρώπους που λαμβάνουν τις αποφάσεις.

Ευτυχώς, πρόκειται περί ενός προβλήματος που σταδιακά επιλύεται, αλλά σε πολλές δημόσιες ομιλίες των παραγόντων της Δημόσιας Διοίκησης, φαίνεται ότι το Ελεύθερο Λογισμικό γίνεται ακόμη αντιληπτό ως κάτι το παράξενο, και έτσι, θεωρούν πως οι αποφάσεις σχετικά με την χρήση του περικλείουν ακόμη κάποιους κινδύνους. Επιπρόσθετα, συναντούμε επίσης και ένα πρόβλημα πολιτικής λήψης αποφάσεων. Το κύριο πλεονέκτημα του Ελεύθερου Λογισμικού για τη Δημόσια Διοίκηση δεν είναι το θέμα του κόστους (αφού το κόστος, σε κάθε περίπτωση, είναι υψηλό, ειδικά όταν συζητάμε για την εισαγωγή του σε έναν μεγάλο αριθμό θέσεων εργασίας), αλλά, σύμφωνα με όσα έχουμε ήδη πει, τα πλεονεκτήματα είναι κυρίως στρατηγικής φύσεως. Συνεπώς, να γιατί η απόφαση εμπίπτει βασικά στη σφαίρα της πολιτικής, παρά στη σφαίρα των τεχνοκρατικών κριτηρίων. Χωρίς την εδραίωση της πολιτικής βούλησης όχι μόνο για μία αλλαγή των συστημάτων λογισμικού, αλλά και της όλης φιλοσοφίας με την οποία συντάσσονται οι προκηρύξεις προμήθειας για αυτά τα συστήματα, θα είναι δύσκολο να σημειωθεί πρόοδος με την απλή εφαρμογή ενός μεμονωμένου προγράμματος Ελεύθερου Λογισμικού στη Δημόσια Διοίκηση.

2) Ελλιπής προσαρμογή των μηχανισμών προκήρυξης διαγωνισμών προμήθειας

Οι μηχανισμοί που χρησιμοποιεί η Δημόσια Διοίκηση μέχρι και σήμερα, ποικίλλουν από τα κλασικά μοντέλα προκήρυξης ενός δημόσιου διαγωνισμού, μέχρι και την ανά είδος κοστολόγηση (*cost itemising*), και έχουν βασικά σχεδιασθεί για την αγορά προϊόντων IT και όχι τόσο για αγορά υπηρεσιών που συσχετίζονται με την παροχή υποστήριξης των προγραμμάτων. Όταν, όμως, χρησιμοποιούμε Ελεύθερο Λογισμικό, συνήθως δεν υφίσταται καν θέμα αγοράς ενός προϊόντος, ή τουλάχιστον πρόκειται για ένα ευτελές τίμημα. Σε αντίθεση, για να μπορούμε να αντλήσουμε τα μέγιστα οφέλη από τις ευκαιρίες που δημιουργούνται με τη χρήση Ελεύθερου Λογισμικού, συμφέρει να προβούμε στην αγορά υπηρεσιών υποστήριξής του. Αυτό σημαίνει πως είναι απαραίτητο, πριν καν προχωρήσουμε στη σοβαρή χρήση Ελεύθερου Λογισμικού, να έχουμε σχεδιάσει τους απαραίτητους γραφειοκρατικούς μηχανισμούς που να διευκολύνουν την προμήθεια των προγραμμάτων και των υπηρεσιών αυτών.

3) Έλλειψη συνολικής στρατηγικής για ανάπτυξη

Συχνά, μία Δημόσια Διοίκηση μπορεί να χρησιμοποιεί Ελεύθερο Λογισμικό διότι απλούστατα είναι ανύπαρκτο ή χαμηλό το κόστος αγοράς του. Σε αυτές τις περιπτώσεις, είναι συνηθισμένο για το εν λόγω λογισμικό να ενσωματωθεί στο υπολογιστικό σύστημα χωρίς κανέναν περαιτέρω σχεδιασμό, και, γενικώς, δίχως μία συνολική στρατηγική για τη χρήση του και την άντληση των μέγιστων δυνατών ωφελειών του Ελεύθερου Λογισμικού. Αυτό οδηγεί στην απώλεια των πλεονεκτημάτων κατά τη διάρκεια της διαδρομής, αφού το όλο θέμα συρρικνώνεται στη χρήση ενός φθηνότερου προϊόντος, ενώ όπως ήδη αναλύσαμε, τα μεγαλύτερα οφέλη είναι, σε γενικές γραμμές, διαφορετικού είδους.

Εάν, επιπρόσθετα, δεν έχει σχεδιασθεί επιμελώς ούτε και η ίδια η διαδικασία μετάβασης, τότε η χρήση Ελεύθερου Λογισμικού μπορεί να οδηγήσει στη δημιουργία σημαντικών κοστολογίων, και όπως θα δούμε σε ορισμένες μεμονωμένες περιπτώσεις, όταν πραγματοποιείται έξω από ένα καλοσχεδιασμένο πλαίσιο, η χρήση του Ελεύθερου Λογισμικού στη Διοίκηση, μπορεί να αποβεί ανεπιτυχής και να προκαλέσει απογοήτευση

4) Σπανιότητα ή και έλλειψη προϊόντων Ελεύθερου Λογισμικού σε ορισμένους τομείς.

Η ανάπτυξη Ελεύθερου Λογισμικού σε οποιαδήποτε οργάνωση μπορεί να αντιμετωπίσει την έλλειψη ελεύθερων ποιοτικών εναλλακτικών επιλογών για κάποια είδη εφαρμογών. Σε αυτές τις περιπτώσεις, η λύση είναι περίπλοκη: το μόνο που μπορούμε να πράξουμε είναι να προωθήσουμε τη δημιουργία ενός τέτοιου ελεύθερου προϊόντος που χρειαζόμαστε. Ευτυχώς, οι Δημόσιοι οργανισμοί βρίσκονται σε πλεονεκτική θέση για να μελετήσουν προσεκτικά κατά πόσο θα τους ενδιέφερε η προώθηση ή ακόμη και η χρηματοδότηση ή συγχρηματοδότηση, για την ανάπτυξη αυτού του προϊόντος. Θα πρέπει να θυμόμαστε ότι ανάμεσα στους σκοπούς τους περιλαμβάνεται συνήθως και η παροχή, για παράδειγμα, προς τους πολίτες τους μίας καλύτερης πρόσβασης στην Κοινωνία της Πληροφορίας, είτε η προώθηση της ανάπτυξης του τοπικού παραγωγικού ιστού. Φυσικά, η δημιουργία πολλών Ελεύθερων προγραμμάτων θα έχει μία θετική επίδραση και στους δύο παραπάνω σκοπούς, από την άποψη ότι θα πρέπει να συνυπολογίσουμε πέραν του ισοζυγίου κόστους/αποτελέσματος, και τα έμμεσα πλεονεκτήματα, που μία τέτοια απόφαση θα επιφέρει.

5) Διαλειτουργικότητα με άλλα προϋπάρχοντα συστήματα

Δεν είναι συχνό να πραγματοποιείται μία πλήρης μετάβαση ολόκληρου του υπολογιστικού συστήματος σε Ελεύθερο Λογισμικό, σε μία μόνο φάση, ταυτοχρόνως. Άρα, είναι σημαντικό να

διασφαλίσουμε ότι εκείνο τα τμήμα που θέλουμε να μετακομίσουμε στο Ελεύθερο Λογισμικό, θα πρέπει να συνεχίσει να λειτουργεί ορθά, μέσα στα πλαίσια όλων των υπολοίπων προγραμμάτων, με τα οποία θα πρέπει να συν-λειτουργεί ομαλά. Αυτό είναι ένα πολύ γνωστό πρόβλημα με οποιαδήποτε περίπτωση μετάβασης (ακόμη κι' αν πρόκειται για ένα διαφορετικό κλειστό, ιδιοταγές προϊόν), αλλά που μπορεί να έχει ιδιαίτερη επίπτωση στην περίπτωση του Ελεύθερου Λογισμικού.

Πάντως, αυτό θα πρέπει πάντα να λαμβάνεται υπ' όψη όταν σχεδιάζει κανείς την μετάβαση σε Ελεύθερο Λογισμικό. Ευτυχώς, έχουμε επίσης την δυνατότητα να παρεμβαίνουμε και να τροποποιούμε το Ελεύθερο Λογισμικό που πρέπει να εγκαταστήσουμε, με τέτοιο τρόπο που να μπορεί να λειτουργήσει ικανοποιητικά και με τα άλλα συστήματα. Εάν, όμως, κάτι τέτοιο απαιτείται, τότε στην κατάσταση του προβλεπόμενου προϋπολογισμού δαπάνης για τη μετάβαση, θα πρέπει να συνυπολογισθεί και αυτό το στοιχείο.

6) Μεταφορά των δεδομένων

Αυτό αποτελεί ένα γενικότερο πρόβλημα, ακόμη και αν πρόκειται για ιδιοταγείς, κλειστές εφαρμογές, σε κάθε περίπτωση μετάβασης σε νέες εφαρμογές που χρησιμοποιούν διαφορετικές μορφές αρχείων. Πράγματι, στην περίπτωση του Ελεύθερου Λογισμικού, το πρόβλημα αυτό συχνά μετριάζεται, μιας και είναι σύνηθες να καταβάλλεται ιδιαίτερη προσπάθεια για να υπάρξει πρόβλεψη να καλύπτονται όσο το δυνατόν περισσότερες προδιαγραφές για τη διαδικασία ανταλλαγής αρχείων και όσο περισσότερες μορφές αρχείων γίνεται. Αλλά, σε κάθε περίπτωση, η μετάπτωση/ μεταφορά των δεδομένων θα πρέπει, φυσιολογικά, να γίνει. Και το κόστος της διαδικασίας αυτής είναι υψηλό. Άρα, στον υπολογισμό του κόστους μίας ενδεχόμενης μετάβασης σε Ελεύθερο Λογισμικό, θα πρέπει να ληφθεί σοβαρά υπ' όψη και αυτός ο συγκεκριμένος παράγοντας.

6.2. Δράσεις των Δημόσιων Οργανισμών στον κόσμο του Ελεύθερου Λογισμικού

Οι Δημόσιοι Οργανισμοί επηρεάζουν τον κόσμο του Ελεύθερου Λογισμικού με τουλάχιστον τρεις τρόπους:

- Μέσω της αγοράς προγραμμάτων και υπηρεσιών που σχετίζονται με αυτά. Οι Δημόσιες Διοικήσεις, που από τη φύση τους είναι μεγάλοι χρήστες λογισμικού, αποτελούν βασικούς παίκτες στην αγορά λογισμικού.
- Μέσω της προώθησης διαφορετικών τρόπων χρήσης (και τρόπου αγοράς) ορισμένων προγραμμάτων, εκ μέρους των μεμονωμένων προσώπων ή και εκ μέρους των εταιρειών. Αυτή η προώθηση επιτυγχάνεται μερικές φορές με την προσφορά οικονομικών κινήτρων (φοροαπαλλαγές, άμεσα κίνητρα, κλπ), μερικές φορές με την πληροφόρηση και την παροχή συμβουλών, και μερικές φορές απλώς διά του “ακολουθήστε το παράδειγμά μου”...
- Μέσω χρηματοδότησης (άμεσης ή έμμεσης) της έρευνας και της ανάπτυξης, και άρα του σχεδιασμού για το μέλλον του λογισμικού.

Σε κάθε έναν από αυτούς τους τομείς, το Ελεύθερο Λογισμικό μπορεί να προσφέρει ιδιαίτερα πλεονεκτήματα (επιπρόσθετα από εκείνα που έχουμε ήδη περιγράψει στο προηγούμενο κεφάλαιο), και τα οποία αφορούν τόσο την ίδια τη Δημόσια Διοίκηση, όσο και την κοινωνία γενικότερα.

6.2.1. Πώς να καλυφθούν οι ανάγκες της Δημόσιας Διοίκησης ;

Οι Δημόσιοι Οργανισμοί είναι μεγάλοι καταναλωτές προϊόντων ΙΤ. Σε ότι αφορά το Λογισμικό, συνήθως αγοράζουν προϊόντα *off-the-shelf* (έτοιμα, από το ράφι), σαν παραμετροποιημένα συστήματα. Από αυτή την άποψη, αποτελούν, ουσιαστικά, μεγάλους αγοραστές, όμοιους με τις μεγάλες εταιρείες, αλλά με τα δικά τους ιδιαίτερα χαρακτηριστικά. Σε πολλούς τομείς, για παράδειγμα, οι αποφάσεις των Δημοσίων οργανισμών για τις αγορές, υποτίθεται ότι θα λάβουν υπ' όψη τους όχι μόνο τις παραμέτρους κόστους/ λειτουργικότητας, αλλά και άλλα στοιχεία, όπως την επίδραση που θα ασκήσει η αγορά στις παροχές κοινωνικού κράτους και εργατικών δικαιωμάτων, είτε και μακροπρόθεσμες εκτιμήσεις, που μπορεί να είναι εξ' ίσου σημαντικές.

Σε κάθε περίπτωση, η πιο συνηθισμένη κατάσταση για το σημερινό λογισμικό “*off-the-shelf*”, είναι να χρησιμοποιεί κανείς τα κορυφαία προϊόντα της αγοράς. Τα ποσά χρημάτων που δαπανούν οι διάφοροι Δήμοι, οι Περιφέρειες και οι εθνικές Κυβερνήσεις, καθώς και υπερεθνικοί οργανισμοί σαν την Ε.Ε., για την αγορά αδειών των Windows, του MS-Office ή άλλων παρόμοιων προϊόντων, είναι πράγματι σημαντικά. Παρ' όλα ταύτα, ελεύθερες λύσεις αρχίζουν ήδη να μπαίνουν στην αγορά. Με αυξητικό ρυθμό, λαμβάνονται υπ' όψη όλο και περισσότερες διάφορες λύσεις που βασίζονται στο Ελεύθερο Λογισμικό, για τους διακομιστές, και προϊόντα σαν το OpenOffice.org, το LibreOffice, ή το Λειτουργικό Σύστημα GNU/ Linux με GNOME ή με KDE, χρησιμοποιούνται όλο και πιο πολύ στους επιτραπέζιους υπολογιστές.

Αλλά τι μπορεί να κερδίσει κανείς από αυτή τη μετάβαση σε Ελεύθερο Λογισμικό; Ακριβώς για να το αναδείξουμε αυτό, ας εστιάσουμε στο ακόλουθο σενάριο. Ας υποθέσουμε ότι με ένα κλάσμα της δαπάνης που προς το παρόν πηγαίνει σε 2-3 κορυφαία ιδιοταγή προϊόντα εκ μέρους όλων των Δημοσίων Διοικήσεων των Ευρωπαϊκών κρατών (ή, ενδεχομένως, εκείνων οποιασδήποτε μεσαίου μεγέθους ανεπτυγμένης χώρας), θα μπορούσαμε να προκηρύξουμε έναν δημόσιο διαγωνισμό, προσκαλώντας μία εταιρεία (ή δύο ή τρεις ή τέσσερις εταιρείες), να βελτιώσουν και να προσαρμόσουν τα ήδη υφιστάμενα ελεύθερα προγράμματα, εις τρόπον ώστε να είναι έτοιμα σε 2-3 χρόνια για μαζική χρήση, τουλάχιστον για κάποιες συνηθισμένες εργασίες (αν δεν το καταφέρνουν ήδη). Ας φαντασθούμε, για παράδειγμα, μία συντονισμένη προσπάθεια, σε πανεθνική ή πανευρωπαϊκή κλίμακα, όπου όλοι οι Δημόσιοι Οργανισμοί θα συμμετέχουν σε μία κοινοπραξία που θα είναι υπεύθυνη για τη διαχείριση αυτών των διαγωνισμών. Σε μία σύντομη χρονική περίοδο, θα δημιουργηθεί μία “εντόπια” βιομηχανία, με εξειδίκευση στην επίτευξη των ζητούμενων βελτιώσεων και προσαρμογών. Και οι διάφοροι Δημόσιοι Οργανισμοί θα μπορούν να διαλέξουν ανάμεσα σε 3-4 ελεύθερες διανομές/ εκδόσεις που θα παράγει αυτή η βιομηχανία. Με στόχο την προώθηση της ανταγωνιστικότητας, η κάθε εταιρεία θα μπορούσε να αμείβεται αναλόγως του αριθμού των Δημοσίων Οργανισμών που επέλεξαν τη δική τους διανομή. Και το συνολικό αποτέλεσμα αυτής της προσπάθειας, εφ' όσον πρόκειται για Ελεύθερο Λογισμικό, θα μπορούσε να καταστεί διαθέσιμο για εταιρείες αλλά και για μεμονωμένους χρήστες, που μπορεί σε πολλές περιπτώσεις να έχουν ανάγκες ανάλογες με εκείνες της Δημόσιας Διοίκησης.

Στη περίπτωση του παραμετροποιημένου λογισμικού, η κλασική διαδικασία προβλέπει την σύναψη σύμβασης με μία εταιρεία, για τα απαραίτητα προγράμματα, με ένα ιδιοταγές κλειστό μοντέλο. Κάθε είδους ανάπτυξη που γίνεται κατόπιν σχετικού αιτήματος του Δημόσιου Οργανισμού, αποτελεί ιδιοκτησία της εταιρείας που το ανέπτυξε. Και συνήθως, η συμβαλλόμενη Δημόσια Αρχή δένεται με τον προμηθευτή της για κάθε τι που σχετίζεται με βελτιώσεις, αναβαθμίσεις και τεχνική υποστήριξη, εν είδει φαύλου κύκλου που καθιστά πολύ δύσκολο τον ανταγωνισμό και επιβραδύνει τη διαδικασία εκσυγχρονισμού των Δημοσίων Οργανισμών. Ακόμη χειρότερο είναι το φαινόμενο όπου το ίδιο πρόγραμμα πωλείται ξανά και ξανά σε διάφορους Δημόσιους φορείς, εφαρμόζοντας

όμως σε όλες αυτές τις περιπτώσεις τα ίδια κόστη που καταλογίσθηκαν αρχικά για την ανάπτυξη του προγράμματος εκ του μηδενός.

Ας εξετάσουμε και πάλι πως θα μπορούσαν τα πράγματα να πάνε διαφορετικά. Μία κοινοπραξία (*consortium*) Δημοσίων Οργανισμών που χρειάζονται ένα ειδικό τύπο παραμετροποιημένου λογισμικού, θα μπορούσαν να ζητήσουν το τελικό προϊόν να είναι Ελεύθερο Λογισμικό. Αυτό θα επέτρεπε σε άλλες Δημόσιες Αρχές να επωφεληθούν από την δουλειά αυτή που έγινε, και θα μπορούσε, μεσοπρόθεσμα, να προσελκύσει το ενδιαφέρον τους για μία συνεργασία με την κοινοπραξία αυτή, με τρόπο που οι ιδιαίτερες απαιτήσεις των δημοσίων Οργανισμών να μπορούν να ληφθούν υπ' όψιν. Καθώς το παραγόμενο λογισμικό θα είναι Ελεύθερο, δεν θα υφίσταται υποχρέωση δέσμευσης, για να υπαχθούν όλες οι βελτιώσεις και προσαρμογές στον ίδιο προμηθευτή, και συνεπώς, ο ανταγωνισμός θα εισέλθει στην αγορά (ενώ, υπό τις σημερινές συνθήκες, πρόκειται, σχεδόν, για εγκλωβισμό).

Σε όλες τις προαναφερθείσες καταστάσεις, το τελικό κόστος για οποιαδήποτε Δημόσια Αρχή που εμπλέκεται, δεν θα υπερβεί ποτέ εκείνο που θα πλήρωνε αν είχε επιλέξει ιδιοταγές λογισμικό.

Μήπως αυτά είναι σενάρια επιστημονικής φαντασίας; Όπως θα δούμε αργότερα, υπάρχουν ήδη κάποιες δειλές πρωτοβουλίες προς κατευθύνσεις παρόμοιες με αυτές που περιγράψαμε. Πέρα και από την προσπάθεια δημιουργίας και συντήρησης μίας βιομηχανίας, στα πλαίσια των Δημοσίων Αρχών που είναι οι αγοραστές, το Ελεύθερο Λογισμικό προσφέρει και πιο εξειδικευμένα πλεονεκτήματα στη σφαίρα του Δημόσιου χώρου. Για παράδειγμα, είναι ο πιο αποτελεσματικός τρόπος για την ανάπτυξη λογισμικού σε γλώσσες εθνικών μειονοτήτων (πράγμα που αποτελεί μία βασική ανησυχία πολλών Δημοσίων Αρχών). Μπορεί επίσης να βοηθήσει πολύ στη διατήρηση της μακροπρόθεσμης στρατηγικής ανεξαρτησίας, αλλά και στη διασφάλιση της προσβασιμότητας των δεδομένων που είναι υπό την κατοχή των Δημοσίων Οργανισμών για πολύ μεγάλα χρονικά διαστήματα. Για όλους αυτούς τους λόγους, οι Δημόσιες Αρχές εκδηλώνουν, ως χρήστες, ένα αυξανόμενο ενδιαφέρον για το Ελεύθερο Λογισμικό.

Μερικές περιπτώσεις που σχετίζονται με Γερμανικές Δημόσιες Υπηρεσίες

Τον Ιούλιο του 2003, είχε εκδοθεί η πρώτη σταθερή έκδοση του Kolab, ένα προϊόν του Kroupware project. Το Kolab είναι ένα ελεύθερο IT σύστημα βοήθειας για ομαδική εργασία (groupware), βασισμένο στο KDE. Ο λόγος που το αναφέρουμε είναι διότι προήλθε, αρχικά, ως αποτέλεσμα μίας προκήρυξης εκ μέρους του Κυβερνητικού Οργανισμού “Bundesamt für Sicherheit in der Informationstechnik” (BSI – μπορεί να μεταφραστεί ως “Ομοσπονδιακό Γραφείο για την ασφάλεια των Πληροφοριών”). Αυτή η προκήρυξη επεδίωκε την επίτευξη μίας λύσης που να λειτουργεί, από τη μία μεριά, με τα Windows και το Outlook, και από την άλλη, με το GNU/Linux και το KDE. Από τις τρεις υποβληθείσες προτάσεις, η κοινή προσφορά τριών εταιρειών, της Erfakon, της Intevation και της Klarälvdalens Datakonsult, κέρδισε την ανάθεση σύμβασης, με την εισήγησή τους να δημιουργήσουν μία ελεύθερη λύση που να βασίζεται εν μέρει σε προϋπάρχον λογισμικό από το KDE project, αλλά το οποίο θα ολοκληρωνόταν και με τα δική τους ελεύθερη ανάπτυξη λογισμικού, πράγμα που οδήγησε στη γέννηση του Kolab.

Τον Μάιο του 2003, ο Δήμος Μονάχου (Γερμανία) ενέκρινε την μετάβαση στο GNU/Linux καθώς και στην ελεύθερη σουίτα γραφείου για όλους τους υπολογιστές desktop που διέθετε, περίπου 14000 εν τω συνόλω. Η απόφαση για να γίνει αυτό δεν ελήφθη με καθαρά οικονομικά κριτήρια: σύμφωνα με όσα δήλωσαν οι υπεύθυνοι αξιωματούχοι, ελήφθησαν υπ' όψιν στρατηγικά και ποιοτικά κριτήρια. Πριν ληφθεί η απόφαση, προηγήθηκε η εκπόνηση μίας εν τω βάθει ανάλυσης, που κατέληξε το συμπέρασμα πως η τελικώς εγκριθείσα λύση (δηλ. το GNU/Linux μαζί με το OpenOffice.org) συγκέντρωσε 6.218 βαθμούς (με άριστα το 10.000), κατ' αντιπαράθεση με τους ελάχιστους πάνω από 5.000 βαθμούς που συγκέντρωσε η “παραδοσιακή” λύση που βασιζόταν σε λογισμικό της Microsoft.

Τον Ιούλιο του 2003, το Ομοσπονδιακό Γραφείο για την ασφάλεια των Πληροφοριών (Koordinierungs-und Beratungsstelle der Bundesregierung für Informationstechnik in der Bundesverwaltung, KBSt), κατόπιν εντολών του Γερμανικού Υπουργ. Εσωτερικών, έδωσε στη δημοσιότητα το έγγραφο “Leitfaden für die Migration von Basissoftwarekomponenten auf Server- und Arbeitsplatzsystemen” [107] (“Οδηγός μετάβασης για τα βασικά στοιχεία λογισμικού των διακομιστών και των σταθμών εργασίας”), που προσέφερε στις Γερμανικές Αρχές ένα σύνολο οδηγιών για το πως να κάνουν τη μετάβαση σε λύσεις βασισμένες στο Ελεύθερο Λογισμικό. Αυτές οι οδηγίες σχεδιάστηκαν για

εκείνους που λαμβάνουν τις αποφάσεις, ώστε να είναι σε θέση να κρίνουν κατά πόσο η μετάβαση σε Ελεύθερο Λογισμικό είναι μία σκόπιμη και κατάλληλη ενέργεια και πώς να την πετύχουν, εφ' όσον το είχαν αποφασίσει.

6.2.2. Προώθηση της ΚτΠ (Κοινωνίας της Πληροφορικής)

Οι Δημόσιες υπηρεσίες δαπανούν πολλούς πόρους για να προσφέρουν κίνητρα και να ενθαρρύνουν τις δαπάνες στον τομέα ΙΤ. Αυτό είναι ένα εξαιρετικό εργαλείο, που θα μπορούσε να βοηθήσει τις νέες τεχνολογίες να διαδοθούν εντός της κοινωνίας. Αλλά είναι συγχρόνως και ένα επικίνδυνο εργαλείο. Για παράδειγμα, μπορεί τελικά να μην είναι και τόσο καλή η ιδέα της προώθησης της χρήσης του Διαδικτύου από την κοινωνία, με μία δική μας σύσταση για έναν συγκεκριμένο φυλλομετρητή, καθώς θα ενθαρρύνουμε έτσι την *de facto* μονοπωλιακή θέση μίας επιχείρησης, και διότι αυτό θα αποβεί μακροπρόθεσμα αρνητικό, για την ίδια την κοινωνία που εμείς προσπαθήσαμε να οφελήσουμε.

Για μία ακόμη φορά, το Ελεύθερο Λογισμικό μπορεί να προσφέρει βοήθεια σε αυτή την κατάσταση. Κατ' αρχάς, είναι ουδέτερο προς τους διάφορους κατασκευαστές, αφού κανείς δεν διαθέτει αποκλειστικότητα για κανένα ελεύθερο πρόγραμμα. Αν μία Δημόσια Αρχή επιθυμεί να προωθήσει μία ολόκληρη οικογένεια ελεύθερων προγραμμάτων, μπορεί να προκηρύξει έναν διαγωνισμό, στον οποίον οποιαδήποτε εταιρεία του χώρου θα μπορεί να συμμετάσχει με την υποβολή μίας προσφοράς, ούτως ώστε η Δημόσια Αρχή να είναι σε θέση, στη συνέχεια, να διανείμει αυτά τα προϊόντα στους πολίτες, μαζί με τις βελτιώσεις ή και τις επηυξημένες λειτουργίες τους, κλπ. Κατά δεύτερο λόγο, μπορεί να προσφέρει βοήθεια σε πολλά θέματα οικονομικής φύσεως. Για παράδειγμα, σε πολλές περιπτώσεις, μπορεί κανείς να δαπανήσει το ίδιο ποσό για την αγορά ενός αριθμού αδειών ιδιοταγών προγραμμάτων, όπως αν αγόραζε ένα ελεύθερο αντίτυπο μαζί με μία σύμβαση για τεχνική υποστήριξη ή και για τις διάφορες τροποποιήσεις που απαιτούνται, είτε ακόμη και διαπραγματευόμενοι με ένα κατασκευαστή λογισμικού τη δυνατότητα να μας παραχωρήσει τα δικαιώματα μετατροπής του προϊόντος του σε Ελεύθερο Λογισμικό.

Σε ένα ξεχωριστό επίπεδο, θα μπορούσαμε να φανταστούμε την αφιέρωση τμήματος των χρημάτων που προορίζονται για τον εξοπλισμό των Σχολείων με υπολογιστές, δημιουργώντας μία διανομή GNU/Linux που να είναι τροποποιημένη σύμφωνα με τις απαιτήσεις των Δημοτικών Σχολείων. Με τα υπόλοιπα χρήματα θα μπορούσαμε να προβούμε στην αγορά συμβολαίων υποστήριξης, για να μπορέσουμε να συντηρούμε τα λογισμικά σε αυτά τα Σχολεία, και έτσι να μην μετατραπουν τα διάφορα προγράμματα σε μία απλή “βιτρίνα-μόστρα”, αλλά αντιθέτως, να διασφαλίσουμε την ύπαρξη υπεύθυνων ανθρώπων που μπορούν πραγματικά να στηρίζουν την καλή τους λειτουργία. Αυτό δεν θα έχει ως αποτέλεσμα μόνον την κάλυψη των διάφορων εκπαιδευτικών αναγκών, αλλά θα δώσει και μία ισχυρή ώθηση για τη δημιουργία μίας αγοράς, συνήθως τοπικών εταιρειών, ικανών να παράσχουν υπηρεσίες υποστήριξης. Και βέβαια, αφήνει το δρόμο για το μέλλον εντελώς ανοιχτό: το λογισμικό δεν θα καταστεί ξεπερασμένο μόλις μέσα σε λίγα χρόνια, που θα σήμαινε να ξαναρχίσουμε όλη τη διαδικασία από την αρχή, αλλά θα μπορεί, αντιθέτως, να αναβαθμίζεται αυξητικά, χρόνο με το χρόνο, διατηρώντας τα πλεονεκτήματα του προγράμματος, με μία παρόμοια επένδυση.

Σημείωση

Οι αναγνώστες που είναι εξοικειωμένοι με τις διάφορες πρωτοβουλίες δημοσίων αρχών σε σχέση με το Ελεύθερο Λογισμικό, θα αναγνωρίσουν ήδη σε αυτό το παράδειγμα, την περίπτωση του gnuLinEx. Προς το τέλος του 2001, η Διοίκηση της Περιφέρειας της Extremadura (Ισπανία) αποφάσισε να χρησιμοποιήσει μία διανομή GNU/Linux, για να εξοπλίσει με υπολογιστές όλα τα Σχολεία της περιοχής της. Για να το πετύχει, χρηματοδότησε τη δημιουργία του gnuLinEx, μίας διανομής GNU/Linux βασισμένης πάνω στο Debian, που ανακοινώθηκε την Άνοιξη του 2002, και εξασφάλισε ότι θα έμπαινε ως προαπαιτούμενο μέσα σε όλες τις προκηρύξεις για προμήθεια υπολογιστών για τα σχολεία. Επίσης, άρχισε και προγράμματα εκπαίδευσης για

τους Δασκάλους, δημιουργώντας εκπαιδευτικό υλικό και αναπτύσσοντας εμπειρία και σε άλλους τομείς. Κατά τα μέσα του 2003, φάνηκε εκ των πραγμάτων ότι αυτή η εμπειρία σημείωσε επιτυχία, καθώς άρχισε να επεκτείνεται και σε θεσμικούς φορείς άλλων Περιφερειών (για παράδειγμα, στην Ανδαλουσία, επίσης στην Ισπανία, μέσω του Guadalinux project).

6.2.3. Προώθηση της έρευνας

Το Ελεύθερο Λογισμικό παρέχει επίσης αξιόλογα πλεονεκτήματα σε ότι αφορά τις θεσμοθετημένες πολιτικές Έρευνας & Ανάπτυξης (R&D policies). Το δημόσιο χρήμα χρησιμοποιείται για την χρηματοδότηση πολυάριθμων σχεδίων ανάπτυξης λογισμικού, από τα οποία η κοινωνία δεν καταλήγει να επωφεληθεί, ούτε καν εμμέσως. Η δημόσια έρευνα και ανάπτυξη, συνήθως χρηματοδοτούν, εν όλω ή εν μέρει, διάφορα projects που θα οδηγήσουν στη δημιουργία λογισμικού, χωρίς όμως στα αλήθεια να τους απασχολεί τι δικαιώματα θα έχει το ευρύ κοινό πάνω σε αυτά τα προϊόντα. Σε πολλές περιπτώσεις τα αποτελέσματα, ελλείπει ενός κατάλληλου σχεδίου εμπορικής προώθησης, απλούστατα αρχειοθετούνται και αφήνονται να μαζεύουν σκόνη. Σε άλλες, οι ίδιοι άνθρωποι που χρηματοδότησαν ένα πρόγραμμα με τα χρήματα των φόρων που πλήρωσαν, καταλήγουν να ξαναπληρώνουν για τα προϊόντα αυτά, εάν αποφασίσουν ότι θέλουν να τα χρησιμοποιήσουν (δεδομένου ότι θα πρέπει να αγοράσουν άδειες χρήσεως).

Το Ελεύθερο Λογισμικό αποτελεί μία ενδιαφέρουσα επιλογή, την οποία οι Αρχές που είναι υπεύθυνες για τις πολιτικές καινοτομίας σε πολλούς Δημόσιους οργανισμούς, αρχίζουν σταδιακά να λαμβάνουν προσεκτικά υπ' όψη. Ιδιαίτερα όταν η έρευνα είναι προ-ανταγωνιστική (pre-competitive), πράγμα πολύ συχνό στην περίπτωση των δημοσίων δαπανών, το γεγονός ότι τα προγράμματα που προκύπτουν είναι ελεύθερα, επιτρέπει συνολικά σε όλη τη βιομηχανία και τον εταιρικό κόσμο (και, κατά συνέπεια, σε όλη την κοινωνία) να επωφελείται τα μέγιστα από το δημόσιο χρήμα που δαπανήθηκε σε Έρευνα και Ανάπτυξη (R&D), στο χώρο του Λογισμικού. Εκεί όπου μία εταιρεία μπορεί μόνο να δει ένα αποτέλεσμα που είναι αδύνατον να πωληθεί, μία άλλη μπορεί να διακρίνει μία επιχειρηματική ευκαιρία.

Κατ' αυτό τον τρόπο, από τη μία μεριά τα αποτελέσματα των προγραμμάτων έρευνας μεγιστοποιούνται, και από την άλλη, αυξάνεται ο ανταγωνισμός ανάμεσα στις εταιρείες που επιθυμούν να κάνουν χρήση των αποτελεσμάτων ενός project, αφού όλες τους θα ανταγωνίζονται πάνω στη βάση των ίδιων προγραμμάτων που προέκυψαν από το project.

Αυτό το μοντέλο δεν είναι κάτι καινούργιο. Σε μεγάλο βαθμό, είναι ακριβώς εκείνο το μοντέλο που επέτρεψε στο διαδίκτυο να αναπτυχθεί. Αν οι Δημόσιες Αρχές απαιτούν για τα αποτελέσματα της έρευνας που έλαβε χώρα με τα δικά τους χρήματα, πως θα πρέπει να διανέμονται υπό τη μορφή Ελεύθερου Λογισμικού, τότε δε θα αποτελέσει έκπληξη αν εμφανιστούν παρόμοιες περιπτώσεις, αλλά σε διαφορετικά επίπεδα. Είτε η έκβαση αυτής της έρευνας θα είναι χαμηλής ποιότητας και άχρηστη (οπότε θα πρέπει να αναθεωρηθεί ο τρόπος με τον οποίον επιλέγονται τα projects προς χρηματοδότηση), είτε η ίδια η δυναμική που θα προκληθεί από το να επιτρέπεται να κυκλοφορούν ελεύθερα, για να μπορεί οποιαδήποτε εταιρεία να τα μετατρέψει σε ένα προϊόν, θα οδηγούσε, απλούστατα, σε απρόβλεπτες εξελίξεις.

6.3. Παραδείγματα νομοθετικών πρωτοβουλιών

Στα επόμενα κεφάλαια θα εξετάσουμε κάποιες ιδιαίτερες νομοθετικές πρωτοβουλίες σχετικά με την προώθηση και χρήση Ελεύθερου Λογισμικού από τις Δημόσιες Αρχές. Εννοείται, βεβαίως, ότι ο κατάλογος που παραθέτουμε δεν είναι πλήρης, και εστιάστηκε σε εκείνες τις πρωτοβουλίες που υπήρξαν, για κάποιους λόγους, πρωτοπόρες (έστω κι' αν δεν εγκρίθηκαν στο τέλος). Όσοι από τους

αναγνώστες ενδιαφέρονται παραπάνω, μπορούν να τον συμπληρώσουν, συμβουλευόμενοι το "GrULIC: Νομοθεσία για τη χρήση Ελεύθερου Λογισμικού από το Κράτος" [133], το οποίο αναφέρει πολλές παρόμοιες περιπτώσεις. Επίσης, σε ένα Παράρτημα (το παράρτημα Δ), περιλαμβάνουμε, για λόγους επεξήγησης, το πλήρες κείμενο, ή τα πιο ενδιαφέροντα κομμάτια, διαφόρων από αυτές τις πρωτοβουλίες.

6.3.1. Σχέδια Νόμου στη Γαλλία

Κατά το 1999 και το 2000, στη Γαλλία, παρουσιάστηκαν δύο σχέδια νόμου σχετικά με το Ελεύθερο Λογισμικό, που αποτέλεσαν μία πρωτοπορία, σε μια μακρά σειρά νομοθετικών αντιπαραθέσεων γύρω από αυτό το θέμα:

- Σχέδιο Νόμου του 1999- 495, συντάχθηκε από τους Laffitte, Trégouet και Cabanel, και κατετέθη στη Γερουσία και στον διακομιστή του Διαδικτύου της Γαλλικής Δημοκρατίας, τον Οκτώβριο του 1999. Μετά από μία διαδικασία δημόσιας διαβούλευσης μέσω του Διαδικτύου (<http://www.senat.fr/consult/loglibre/index.htm>) [102] που διήρκεσε επί δύο μήνες, το αρχικό Σχέδιο Νόμου τροποποιήθηκε. Το αποτέλεσμα ήταν το Σχέδιο Νόμου υπ. αριθμ. 2000- 117 (Laffitte, Trégouet και Cabanel, Πρόταση Νόμου υπ. αριθμ. 117, Γερουσία της Γαλλικής Δημοκρατίας, 2000) [162], το οποίο συνηγορούσε υπέρ της υποχρεωτικής φύσης της χρήσης του Ελεύθερου Λογισμικού από τις Δημόσιες Αρχές, προβλέποντας εξαιρέσεις, καθώς και περιόδους προσαρμογής για εκείνες τις περιπτώσεις όπου κάτι τέτοιο δεν ήταν ακόμη τεχνικά εφικτό, μέσα σε ένα γενικότερο πλαίσιο επέκτασης της χρήσης του Διαδικτύου και του Ελεύθερου Λογισμικού μέσα στις Γαλλικές Δημόσιες Υπηρεσίες.
- Τον Απρίλιο του 2000, τα μέλη του Κοινοβουλίου Jean-Yves Le Déaut, Christian Paul και ο Pierre Cohen, πρότειναν έναν νέο Νόμο του οποίου ο σκοπός ήταν παρόμοιος με εκείνον του Σχεδίου Νόμου των Laffitte, Trégouet και Cabanel: να ενισχύσει τις ελευθερίες και την ασφάλεια των καταναλωτών, πέραν από το να βελτιώσει την ισότητα των δικαιωμάτων στην ΚτΠ (Κοινωνία της Πληροφορικής). Όμως, αντίθετα με το Σχέδιο Νόμου των Laffitte, Trégouet και Cabanel, αυτή η δεύτερη πρόταση δεν καθιστούσε υποχρεωτική την χρήση του Ελεύθερου Λογισμικού για τον Δημόσιο Τομέα. Αυτό το σχέδιο νόμου επικεντρώθηκε στο ότι το λογισμικό που χρησιμοποιείται στον Δημόσιο Τομέα θα πρέπει να έχει ελεύθερο τον πηγαίο κώδικά του, χωρίς όμως να καθιστά υποχρεωτική την αναδιανομή του κάτω από ελεύθερες άδειες. Για να πετύχουν τους στόχους τους, οι νομοθέτες προσπάθησαν να εξασφαλίσουν το “δικαίωμα στη διαλειτουργικότητα” του λογισμικού, παρέχοντας μηχανισμούς που έκαναν πράξη την αρχή της διαλειτουργικότητας που προβλέπει η Ευρωπαϊκή Οδηγία σχετική με την νομική προστασία των προγραμμάτων για υπολογιστές (Οδηγίες Ευρωπαϊκού Συμβουλίου 91/250/ EEC, της 14ης Μαΐου του 1991, σχετική με την νομική προστασία των προγραμμάτων για υπολογιστές, 1991) [111].

Κανένα από τα δύο Γαλλικά σχέδια νόμου δεν προκρίθηκε για να μετατραπεί σε Νόμο, αλλά και τα δύο χρησίμευσαν ως έμπνευση για πολλές πρωτοβουλίες που ακολούθησαν σε παγκόσμιο επίπεδο, πράγμα που εξηγεί και το γιατί είναι ιδιαίτερα ενδιαφέρουσα η μελέτη τους. Η δεύτερη πρόταση (που κατατέθηκε από τους Le Déaut, Paul και Cohen) επεδίωκε τη συμβατότητα και την διαλειτουργικότητα του λογισμικού, δίνοντας έμφαση στην ελεύθερη διάθεση του Πηγαίου Κώδικα για όσα προγράμματα χρησιμοποιεί η Δημόσια Διοίκηση. Από την άλλη πλευρά, δεν απαιτούσε τα προγράμματα που θα παράγονταν να είναι και αυτά Ελεύθερα Λογισμικά, υπό την έννοια του να

διανέμονται κάτω από Άδειες που να εγγυώνται την ελευθερία τροποποίησης, χρήσης και αναδιανομής της εφαρμογής.

Αργότερα (στο Κεφάλαιο Δ.1 και στην παράγραφο Δ.2 του Παραρτήματος Δ), παραθέτουμε σχεδόν εξ' ολοκλήρου τα άρθρα και τα επεξηγηματικά υπομνήματα και των δύο σχεδίων νόμου. Τα επεξηγηματικά υπομνήματα είναι ιδιαίτερος ενδιαφέροντα, καθώς τονίζουν τα προβλήματα που απειλούν, στις μέρες μας, τις Δημόσιες Διοικήσεις, για τη χρήση λογισμικού, γενικά.

6.3.2. Σχέδια Νόμου στη Βραζιλία

Το 1999, ο βουλευτής Walter Pinheiro παρουσίασε ένα Σχέδιο Νόμου για το Ελεύθερο Λογισμικό, στην Ομοσπονδιακή Βουλή της Βραζιλίας (Proposição pl-2269/1999. Dispõe sobre a utilização de programas abertos pelos entes de direito público e de direito privado sob controle acionário da administração pública, Βραζιλιανό Κοινοβούλιο, Δεκέμβριος 1999) [185]. Αυτό το project αφορούσε τη χρήση Ελεύθερου Λογισμικού στη Δημόσια Διοίκηση και στις ιδιωτικές επιχειρήσεις, με το κράτος στο ρόλο του μετόχου πλειοψηφίας.

Συστήνει τη χρήση Ελεύθερου Λογισμικού από αυτούς τους Δημόσιους Οργανισμούς, χωρίς περιορισμούς σε ότι αφορά το να δανείζει κανείς το πρόγραμμα σε κάποιον άλλον, να μπορεί να το τροποποιεί ή και να το αναδιανέμει. Τα άρθρα του Νόμου περιγράφουν λεπτομερώς πως ορίζεται το Ελεύθερο Λογισμικό και πως θα πρέπει να είναι ακριβώς οι Άδειες που το συνοδεύουν. Οι ορισμοί αυτοί συμπίπτουν με τον κλασικό ορισμό περί Ελεύθερου Λογισμικού, όπως τον ορίζει το Έργο GNU. Το επεξηγηματικό υπόμνημα κάνει μία επισκόπηση της ιστορίας του έργου GNU, αναλύοντας τα πλεονεκτήματά του και τα όσα πέτυχε. Αναφέρεται, επίσης, στην τωρινή κατάσταση του Ελεύθερου Λογισμικού, χρησιμοποιώντας ως παράδειγμα το Λειτουργικό Σύστημα GNU/Linux.

Μία από τις πιο ενδιαφέρουσες παραγράφους του πρώτου άρθρου, ορίζει με μεγάλη σαφήνεια τη σφαίρα εντός της οποίας προτείνεται η χρήση Ελεύθερου Λογισμικού (κρατώντας κατά νου ότι ο ορισμός που παρέχεται στα επόμενα άρθρα περί "Ανοικτού προγράμματος" είναι, όπως έχουμε ήδη αναφέρει, ο ίδιος όπως και για το Ελεύθερο Λογισμικό):

"Η Δημόσια Διοίκηση σε όλα τα επίπεδα, οι Αρχές της Δημοκρατίας, το Κράτος, καθώς και οι μικτές επιχειρήσεις με συμμετοχή Δημοσίου-Ιδιωτικού τομέως, οι Δημόσιες επιχειρήσεις, αλλά και κάθε άλλη Αρχή Δημοσίου ή Ιδιωτικού Δικαίου που ελέγχεται από το Βραζιλιανό κράτος, υποχρεούνται να χρησιμοποιούν κατά προτεραιότητα, στα υπολογιστικά τους συστήματα και στον εξοπλισμό τους, ανοικτά προγράμματα, ελεύθερα από κάθε ιδιοταγή περιορισμό σχετικά με την παραχώρησή τους σε άλλους, την τροποποίησή τους, και την αναδιανομή τους."

6.3.3. Σχέδια Νόμου στο Περού

Στο Περού, διάφορα σχέδια νόμου προτάθηκαν για τη χρήση Ελεύθερου Λογισμικού από την Δημόσια Διοίκηση ("GNU Perú. Draft laws on free software in the public administration of the Peruvian government", Γερουσία της Δημοκρατίας) [184].

Το πρώτο και το πλέον διάσημο είχε κατατεθεί από τον Γερουσιαστή Edgar Villanueva Núñez, τον Δεκέμβρη του 2001 (Σχέδιο Νόμου περί Ελεύθερου Λογισμικού, υπ' αριθμ. 1609, Δεκ. 2001) [222]. Ορίζει το Ελεύθερο Λογισμικό σύμφωνα με τον κλασικό ορισμό των τεσσάρων ελευθεριών (προσθέτοντας, ίσως, περισσότερη νομική ακρίβεια, με ένα ορισμό που διευκρινίζει έξι χαρακτηριστικά που ορίζουν ένα ελεύθερο πρόγραμμα) και προτείνει την αποκλειστική του χρήση από την Περουβιανή Κυβέρνηση:

"Άρθρο 2. Οι εκτελεστικές, νομοθετικές και δικαστικές αρχές, οι αποκεντρωμένοι οργανισμοί και οι εταιρείες

όπου το Κράτος διατηρεί την πλειοψηφία των μετοχών, θα χρησιμοποιούν αποκλειστικά ελεύθερα προγράμματα ή λογισμικό, για τους υπολογιστές και τον εξοπλισμό τους."

Παρ' όλα ταύτα, αργότερα, τα άρθρα 4 και 5, περιέλαβαν ορισμένες εξαιρέσεις σε αυτόν τον κανόνα. Στην εποχή του, αυτό το σχέδιο νόμου είχε έναν ευρύτερο αντίκτυπο. Από τη μία, υπήρξε η πρώτη φορά που προτάθηκε η αποκλειστική χρήση ελεύθερου λογισμικού από μία δημόσια διοίκηση. Αλλά ο σημαντικότερος αντίκτυπος αυτής της καινοτομίας ήταν η ανταλλαγή επιστολών ανάμεσα στο Γερουσιαστή Villanueva και την αντιπροσωπεία της Microsoft στο Peru, η οποία διατύπωνε κάποιους ισχυρισμούς και κατηγορίες κατά του σχεδίου νόμου. Εξ' άλλου, αυτό το σχέδιο νόμου είναι σημαντικό σε σχέση και με την θέση που πήρε το Αμερικανικό Προξενείο, που έφθασε μέχρι το σημείο να στείλει, μέσω επισήμων διαύλων επικοινωνίας, μία προειδοποίηση (επισυνάπτοντας μία αναφορά που είχε συντάξει η ίδια η Microsoft) προς την Περουβιανή Γερουσία, εκφράζοντας τις "ανησυχίες του σχετικά με τις πρόσφατες προτάσεις εκ μέρους της Γερουσίας της Δημοκρατίας, για να περιορισθούν οι προμήθειες της Περουβιανής Κυβερνήσεως μόνον στο Ανοικτό Λογισμικό ή το Ελεύθερο Λογισμικό" ("Letter to the president of the Congress of the Republic", 2002) [147]. Ανάμεσα στα άλλα, οι ισχυρισμοί τόσο της Microsoft όσο και της Αμερικανικής Κυβερνήσεως, προσπαθούσαν να αποδείξουν ότι το Σχέδιο Νόμου για το Ελεύθερο Λογισμικό θα διαφοροποιούσε ανάμεσα στις διάφορες εταιρείες, καθιστώντας αδύνατες τις επενδύσεις που χρειαζόταν, για τη δημιουργία μίας εθνικής βιομηχανίας κατασκευής λογισμικού.

Ο Γερουσιαστής Villanueva ανταπάντησε με το επιχείρημα ότι το σχέδιο νόμου δεν έκανε καμία διάκριση και κατά κανέναν τρόπο, υπέρ ή κατά οποιασδήποτε συγκεκριμένης εταιρείας, αφού δεν περιέγραφε ποιος θα μπορούσε να είναι ο προμηθευτής της Δημόσιας Διοίκησης, αλλά μάλλον διευκρίνιζε το πως (υπό ποιες συνθήκες) θα μπορούσε να γίνει η προμήθεια του λογισμικού. Αυτό το σκεπτικό είναι πολύ ξεκάθαρο για να μπορέσουμε να κατανοήσουμε πως η προώθηση του Ελεύθερου Λογισμικού εκ μέρους της Δημόσιας Διοίκησης δεν παρενοχλεί, κατά κανένα τρόπο, τον ελεύθερο ανταγωνισμό μεταξύ των διαφόρων προμηθευτών.

Αργότερα, οι Περουβιανοί Γερουσιαστές Edgar Villanueva Núñez και ο Jacques Rodrich Ackerman παρουσίασαν ένα νέο σχέδιο νόμου, υπ' αριθμ. 2485, στις 8 Απριλίου 2002 (Σχέδιο Νόμου για τη χρήση Ελεύθερου Λογισμικού στη Δημόσια Διοίκηση υπ' αριθμ. 2485, 2002) [223], το οποίο κατά τον Αύγουστο του 2003 βρισκόταν ακόμη υπό Κοινοβουλευτική συζήτηση. Αυτό το Σχέδιο Νόμου ήταν μία εξέλιξη του Σχεδίου Νόμου 1609 [222], από το οποίο και αντλεί διάφορα σχόλια και επιφέρει πολυάριθμες βελτιώσεις, και μπορεί να θεωρηθεί ως ένα καλό παράδειγμα Σχεδίου Νόμου που προτείνει την αποκλειστική χρήση του Ελεύθερου Λογισμικού στους Δημόσιους Οργανισμούς, εκτός από μερικές εξαιρετικές περιπτώσεις. Με δεδομένη τη σημασία του, συμπεριλάβαμε το πλήρες κείμενο (κεφάλαιο Δ.3, του Παραρτήματος Δ). Το επεξηγηματικό του υπόμνημα αποτελεί μία καλή περίληψη όλων εκείνων των χαρακτηριστικών που θα πρέπει να διαθέτει το λογισμικό που χρησιμοποιείται από έναν δημόσιο οργανισμό και εξηγεί πως το Ελεύθερο Λογισμικό συμμορφώνεται κατά πολύ καλύτερο τρόπο με αυτά τα χαρακτηριστικά, από ότι το ιδιοταγές λογισμικό.

6.3.4. Σχέδια Νόμου στην Ισπανία

Στην Ισπανία έχουν προκύψει διάφορες πρωτοβουλίες για το Ελεύθερο Λογισμικό. Στη συνέχεια, παραθέτουμε μερικές από αυτές:

- Διάταγμα περί Μέτρων για την προώθηση της Κοινωνίας της Γνώσης στην Andalucía One: η κίνηση αυτή της Ανδαλουσίας είναι, αναμφίβολα, μία από τις πιο σημαντικές νομοθετικές

πρωτοβουλίες στην Ισπανία, καθότι τελικά ετέθη σε εφαρμογή.

Το Διάταγμα περί Μέτρων για την προώθηση της Κοινωνίας της Γνώσης στην Andalucia (Διάταγμα 72/2003, στις 18/03/2003, περί Μέτρων για την προώθηση της Κοινωνίας της Γνώσης στην Andalucia) [99] ασχολείται με τη χρήση Ελεύθερου Λογισμικού, κυρίως (αλλά όχι μόνο) στα πλαίσια της Εκπαίδευσης.

Μεταξύ άλλων, προωθεί την κατά προτίμηση χρήση του Ελεύθερου Λογισμικού στα δημόσια Παιδαγωγικά ιδρύματα, θεσπίζοντας την υποχρέωση για τον εξοπλισμό όλων αυτών των οργανισμών να είναι συμβατός με τα Ελεύθερα Λειτουργικά Συστήματα, και παρομοίως και για τα Διοικητικά κέντρα της Περιφέρειας που παρέχουν πρόσβαση στο Διαδίκτυο.

- Σχέδιο Νόμου για το Ελεύθερο Λογισμικό στα πλαίσια της Δημόσιας Διοίκησης της Καταλωνίας. Άλλες κοινότητες συζήτησαν ενδεχομένως και για εφαρμογή ακόμη πιο φιλόδοξων προτάσεων, χωρίς όμως να πετύχουν την πλειοψηφία που χρειαζόταν. Η πιο διάσημη ανάμεσα σε αυτές ήταν, πιθανώς, εκείνη που συζητήθηκε στο Καταλανικό Κοινοβούλιο (Proposició de llei de programari lliure en el marc de l'Administració pública de Catalunya, 2002) [221], που ήταν πολύ όμοια με εκείνη που το ίδιο κόμμα (Esquerra Republicana de Catalunya) είχε παρουσιάσει στη Γερουσία των Αντιπροσώπων, περί της οποίας θα συζητήσουμε στη συνέχεια. Αυτή η πρόταση ήταν ανεπιτυχής, όταν υπεβλήθη προς ψηφοφορία.

- Σχέδιο Νόμου του Puigcercós Boixassa στη Γερουσία των Αντιπροσώπων. Υπήρξε επίσης μία πρωτοβουλία στη Γερουσία των Αντιπροσώπων, που είχε προταθεί από τον Joan Puigcercós Boixassa (του κόμματος Esquerra Republicana de Catalunya) (Σχέδιο Νόμου περί Μέτρων για την ενσωμάτωση του Ελεύθερου Λογισμικού στην Κρατική Διοίκηση, 2002) [188]. Αυτή η συγκεκριμένη πρωτοβουλία πρότεινε την κατά προτίμηση χρήση του Ελεύθερου Λογισμικού εκ μέρους της Δημόσιας Διοίκησης και, κατ' αυτή την έννοια, ήταν παρόμοια με άλλες πρωτοβουλίες που είχαν τον ίδιο στόχο. Πάντως, αυτή εδώ παρουσίαζε την ενδιαφέρουσα ιδιαιτερότητα να δίνει έμφαση στην διαθεσιμότητα Ελεύθερων προγραμμάτων που να είναι προσαρμοσμένα στις διάφορες κατά τόπους αποδεκτές ως επίσημες γλώσσες (τουλάχιστον σε εκείνες τις αυτόνομες επαρχίες όπου αυτές χρησιμοποιούνται). Ούτε και αυτή η πρωτοβουλία έγινε αποδεκτή κατά την κοινοβουλευτική ψηφοφορία.

7. Η Τεχνολογία του Ελεύθερου Λογισμικού

"Ο καλύτερος τρόπος για να έχεις μία καλή ιδέα, είναι να κατεβάζεις πολλές από αυτές".
Linus Pauling

Στα προηγούμενα κεφάλαια δείξαμε γιατί η πρόκληση του Ελεύθερου Λογισμικού δεν είναι σαν εκείνη την περίπτωση όπου έχεις έναν ανταγωνιστή που δημιουργεί λογισμικό πολύ ταχύτερα, πολύ φθηνότερα και καλύτερης ποιότητας: το Ελεύθερο Λογισμικό διαφοροποιείται από το "παραδοσιακό" λογισμικό κατά πολύ ριζικότερους τρόπους, αρχίζοντας από τις φιλοσοφικές του προϋποθέσεις, τις αιτίες και τα κίνητρα, συνεχίζοντας με τις νέες αγορές και τα οικονομικά μοντέλα και τελειώνοντας με έναν πολύ διαφορετικό τρόπο παραγωγής του ίδιου του λογισμικού. Η Τεχνολογία λογισμικού (software engineering) δεν μπορεί να μείνει ανεπηρέαστη από όλους τους παραπάνω παράγοντες. Έτσι, λοιπόν, σε λίγο περισσότερο από δέκα χρόνια έρευνας γύρω από τους τρόπους με τους οποίους παράγεται το Ελεύθερο Λογισμικό, η προσοχή των ερευνητών εστιάσθηκε σε αυτά τα θέματα σε πολύ μεγαλύτερο βάθος.

Σε αυτό το κεφάλαιο θα αναλύσουμε τις σημαντικότερες μελέτες και τα στοιχεία που απέδωσαν, με στόχο να προσφέρουμε στον αναγνώστη μία συνολική εικόνα της τρέχουσας κατάστασης και των μελλοντικών προοπτικών για εκείνο που θα αποκαλούμε Τεχνολογία του Ελεύθερου Λογισμικού.

7.1. Εισαγωγή

Αν και το Ελεύθερο Λογισμικό αναπτύσσεται εδώ και αρκετές δεκαετίες, μόνο κατά τα τελευταία χρόνια αρχίσαμε να εστιάζουμε την προσοχή μας στα μοντέλα ανάπτυξής του και στις διαδικασίες που το χαρακτηρίζουν, από τη σκοπιά της Τεχνολογίας Λογισμικού. Με τον ίδιο τρόπο που δεν υπάρχει ένα ενιαίο μοντέλο για την ανάπτυξη του ιδιοταγούς λογισμικού, έτσι δεν υφίσταται ούτε και ένα ενιαίο μοντέλο για την ανάπτυξη του Ελεύθερου Λογισμικού, αλλά, έστω κι' έτσι, μπορούμε να εντοπίσουμε ενδιαφέροντα χαρακτηριστικά που είναι κοινά για τα περισσότερα projects υπό μελέτη και που απορρέουν από τις ίδιες τις ιδιότητες των ελεύθερων λογισμικών.

⁽⁵⁾ Το άρθρο "The ecology of open source software development" ("Η οικολογία της ανάπτυξης λογισμικού Ανοικτού Κώδικα"), των Kieran Healy και Alan Schussman, 2003 [140], δείχνει την μεγάλη ποικιλία των projects καθώς και τις διαφορές τους ως προς τον αριθμό developers, τη χρήση των εργαλείων και των downloads.

Το 1997, ο Eric S. Raymond δημοσίευσε το πρώτο ευρέως διαδεδομένο άρθρο "Ο καθεδρικός και το παζάρι, Στοχασμοί γύρω από το Linux και τον Ανοικτό Κώδικα, από έναν τυχαίο επαναστάτη" (The cathedral and the bazaar, Musings on Linux and open source by an accidental revolutionary, Εκδ. O'Reilly, <http://www.ora.com>, 2001) [192], που περιγράφει μερικά από τα χαρακτηριστικά των μοντέλων ανάπτυξης του Ελεύθερου Λογισμικού, δίνοντας ιδιαίτερη έμφαση σε εκείνο που ξεχωρίζει αυτά τα μοντέλα από εκείνα του ιδιοταγούς λογισμικού. Από τότε, αυτό το άρθρο έγινε ένα από τα πιο διάσημα (αλλά και από αυτά που συγκέντρωσαν τις περισσότερες κριτικές) στον κόσμο του Ελεύθερου Λογισμικού, και μέχρι ενός σημείου, απετέλεσε το σημείο έναρξης της γενικότερης ανάπτυξης της Τεχνολογίας του Ελεύθερου Λογισμικού.

7.2. Ο Καθεδρικός και το Παζάρι

Ο Raymond κάνει μία αναλογία ανάμεσα στον τρόπο που χτίζονταν οι μεσαιωνικοί καθεδρικοί ναοί και τον κλασικό τρόπο παραγωγής λογισμικού. Επιχειρηματολογώντας ότι και στις δύο περιπτώσεις υπάρχει μία ξεκάθαρη κατανομή αρμοδιοτήτων και λειτουργιών, με έμφαση στην ύπαρξη ενός σχεδιαστή, που επιβλέπει τα πάντα και οφείλει να έχει τον έλεγχο της ανάπτυξης για όλη τη δραστηριότητα, ανά πάσα στιγμή.

Συγχρόνως, ο σχεδιασμός ελέγχεται αυστηρά, οδηγώντας στη δημιουργία λεπτομερειακών διαδικασιών, όπου ο κάθε συμμετέχων στη δραστηριότητα έχει, ιδεανικά, έναν πολύ καλά καθορισμένο ρόλο.

Εκείνο που ο Raymond παίρνει ως ένα μοντέλο για το χτίσιμο των καθεδρικών, όχι μόνο αφήνει χώρο για τις βαριές διαδικασίες που μπορούμε να βρούμε στη βιομηχανία λογισμικού (το κλασικό μοντέλο του καταρράκτη, οι διάφορες πτυχές του Rational Unified Process, κλπ.), αλλά και για τα projects Ελεύθερου Λογισμικού σαν το GNU και το NetBSD. Για τον Raymond, αυτά τα projects είναι ιδιαίτερα συγκεντρωτικά, αφού μόνο λίγοι άνθρωποι είναι υπεύθυνοι για τη σχεδίαση και την υλοποίηση του λογισμικού. Τα καθήκοντα που εκτελούν αυτοί οι άνθρωποι, επιπροσθέτως των λειτουργιών τους, είναι σαφώς καθορισμένα, και οποιοσδήποτε επιθυμεί να συμμετάσχει σε μία ομάδα ανάπτυξης, πρέπει πρώτα να του αποδοθούν συγκεκριμένες αρμοδιότητες και καθήκοντα, σύμφωνα με τις απαιτήσεις του project. Από την άλλη μεριά, η απόδοση στην κυκλοφορία αυτού του είδους των προγραμμάτων συντονίζεται χρονικά, με ένα αρκετά αυστηρό πρόγραμμα. Αυτό συνεπάγεται λίγες εκδόσεις του λογισμικού, με μεγάλους κύκλους, που συνίστανται σε πολλά ενδιάμεσα στάδια.

Το αντίθετο μοντέλο από εκείνο του Καθεδρικού, είναι το μοντέλο του Παζαριού. Σύμφωνα με τον

Raymond, μερικά από τα προγράμματα Ελεύθερου Λογισμικού, ιδιαίτερα ο πυρήνας Linux, αναπτύχθηκαν σύμφωνα με ένα σχήμα που μοιάζει με εκείνο ενός ανατολίτικου παζαριού. Σε ένα παζάρι δεν υπάρχει μία ανώτερη αρχή που να ελέγχει τις αναπτυσσόμενες διαδικασίες ή που να προγραμματίζει με αυστηρότητα τι πρέπει να γίνει. Συγχρόνως, οι ρόλοι των συμμετεχόντων μπορούν να αλλάζουν συνεχώς (οι πωλητές μπορεί να μετατραπούν σε πελάτες) και χωρίς καμία προς τα έξω ένδειξη.

Εκείνο, όμως, που είναι καινοτόμο στο "The cathedral and the bazaar", είναι η περιγραφή της διαδικασίας με την οποία το Linux έγινε επιτυχία. Πρόκειται για μία λίστα "καλών πρακτικών" που θα μας επιτρέψουν να αντλήσουμε τα μέγιστα δυνατά οφέλη από τις ευκαιρίες που προσφέρει η ελεύθερη διαθεσιμότητα του Πηγαίου Κώδικα, και διαδραστικότητας, μέσω της χρήσης τηλεματικών συστημάτων και εργαλείων.

Ένα project Ελεύθερου Λογισμικού μοιάζει σαν αποτέλεσμα μίας καθαρά ατομικής δράσης. Η αιτία μπορεί να εντοπισθεί σε ένα developer που ανακαλύπτει ότι η ικανότητά του για την επίλυση ενός προβλήματος είναι περιορισμένη. Ο developer χρειάζεται να διαθέτει αρκετή γνώση για να αρχίσει, τουλάχιστον, να το λύνει. Όταν θα έχει πετύχει να δημιουργήσει κάτι που να μπορεί να χρησιμοποιηθεί, να έχει κάποια λειτουργικότητα, να είναι απλό, και ει δυνατόν, να είναι και καλοσχεδιασμένο ή καλογραμμένο, τότε το καλύτερο που έχει να κάνει είναι να μοιραστεί αυτή τη λύση με τον κόσμο του Ελεύθερου Λογισμικού. Πρόκειται για εκείνο που είναι γνωστό σαν "πρώιμη έκδοση" (release early), που βοηθά στην προσέλευση της προσοχής άλλων ανθρώπων (συνήθως developers), που αντιμετωπίζουν επίσης το ίδιο πρόβλημα. Άρα, μπορεί να ενδιαφέρονται για τη λύση αυτή.

Μία από τις βασικές αρχές αυτού του μοντέλου ανάπτυξης είναι να θεωρεί τους χρήστες σαν συνδημιουργούς (co-developers). Πρέπει να τους μεταχειρίζεται κανείς με φροντίδα, όχι διότι μπορούν απλώς να λειτουργήσουν σαν ζωντανή "στόμα-με-στόμα" διαφήμιση, αλλά και διότι αυτοί είναι που θα φέρουν εις πέρας ένα από τα πλέον κοστοβόρα καθήκοντα που υπάρχουν, στην όλη διαδικασία δημιουργίας λογισμικού: τη δοκιμή. Σε αντίθεση με την από κοινού ανάπτυξη, που είναι δύσκολο να κλιμακωθεί, η αποσφαλμάτωση και οι δοκιμές (tests) έχουν την ιδιότητα ότι μπορούν να εκτελεσθούν από μία ευρεία βάση δοκιμαστών και με παράλληλο τρόπο. Ο χρήστης είναι εκείνος που θα πάρει το λογισμικό και θα το δοκιμάσει στο δικό του μηχάνημα, σε συγκεκριμένες συνθήκες (μία αρχιτεκτονική, ορισμένα εργαλεία, κλπ). Αυτή η διαδικασία, πολλαπλασιαζόμενη επί έναν μεγάλο αριθμό διαφορετικών αρχιτεκτονικών και περιβαλλόντων, όσοι είναι και οι δοκιμαστές-χρήστες, θα καθιστούσε αναγκαίο για την ομάδα ανάπτυξης να καταβάλλει μία τεράστια προσπάθεια, εάν όλο αυτό το βάρος έπεφτε στους ώμους της.

Αν μεταχειρισθούμε τους χρήστες σαν συνδημιουργούς, τότε θα μπορούσε κάποιος από αυτούς να εντοπίσει ένα σφάλμα (bug) και να το λύσει, στέλνοντας μία προσωρινή λύση (patch) στους δημιουργούς του project, έτσι ώστε το πρόβλημα να εξαλειφθεί στην επόμενη έκδοση. Μπορεί, επίσης, αυτός που θα καταφέρει να κατανοήσει το bug και ενδεχομένως να το διορθώσει, να είναι κάποιος άλλος από εκείνον που το ανακάλυψε. Σε κάθε περίπτωση, όλες αυτές οι καταστάσεις αποβαίνουν εις όφελος της ανάπτυξης του Ελεύθερου Λογισμικού, που σημαίνει ότι είναι ευεργετικό να μπει κανείς σε αυτή τη δυναμική.

Η κατάσταση γίνεται αποτελεσματικότερη όταν υπάρχουν συχνές εκδόσεις, αφού το κίνητρο για τον εντοπισμό, τη γνωστοποίηση και τη διόρθωση των σφαλμάτων (bugs), είναι υψηλό, μιας και υποτίθεται ότι θα επιδιορθωθούν αμέσως. Επιπλέον, επιτυγχάνονται δευτερογενή οφέλη, όπως το ότι η συχνή ενσωμάτωση- ιδανικά μία ή και περισσότερες φορές την ημέρα- δεν απαιτεί την ύπαρξη μίας τελικής φάσης ενσωμάτωσης όλων των υποενοτήτων που συνθέτουν το πρόγραμμα.

Αυτό έχει ονομασθεί Συχνή Έκδοση (release often) και επιτρέπει έναν μεγάλο βαθμό διάρθρωσης σε υποενότητες (modularity) (Alessandro Narduzzo και Alessandro Rossi, "Modularity in action: GNU/Linux and free/ open source software development model unleashed", May 2003) [176], ενώ ταυτόχρονα μεγιστοποιεί την επίδραση της διαφήμισης που συνεπάγεται η κυκλοφορία της εκάστοτε πιο πρόσφατης έκδοσης του λογισμικού.

Σημείωση

Η διαχείριση της νέας έκδοσης εξαρτάται, λογικά, από το μέγεθος του project, αφού τα προβλήματα που πρέπει να αντιμετωπισθούν δεν μπορεί να είναι τα ίδια όταν η ομάδα ανάπτυξης έχει δύο μέλη, σε σχέση με το να διαθέτει εκατοντάδες μέλη. Ενώ, σε γενικές γραμμές, στην περίπτωση των μικρών projects, η όλη διαδικασία είναι λίγο-πολύ χαλαρή και ανεπίσημη, η διαχείριση των νέων εκδόσεων των μεγάλων projects τείνει να ακολουθεί αυστηρά προσδιορισμένες διαδικασίες, πράγμα που δεν αποκλείει την ύπαρξη ενός ορισμένου βαθμού πολυπλοκότητας. Υπάρχει μία δημοσίευση που έχει ως τίτλο "Release management within open source projects" (Justin R. Ehrenkrantz, 2003) [110] που περιγράφει με λεπτομέρειες όλη την αλληλουχία που ακολουθήθηκε στην περίπτωση του Apache web server, του πυρήνα Linux και του συστήματος Subversion.

Με την λεγόμενη "Συχνή έκδοση" (release often), μπορεί να παρατηρηθεί ο εκφοβισμός των χρηστών που έχουν ως προτεραιότητα τη σταθερότητα του συστήματος και δεν τους ενδιαφέρει τόσο η ταχύτητα ανάπτυξης. Για να προληφθεί το φαινόμενο αυτό, μερικά projects Ελεύθερου Λογισμικού διαθέτουν διάφορα τμήματα ανάπτυξης που τρέχουν ταυτόχρονα. Η πιο διάσημη περίπτωση είναι εκείνη με τον πυρήνα του Linux, που ιστορικά προσφέρει μία έκδοση σε εκείνους που εκτιμούν την αξιοπιστία του, και μία διαφορετική, ασταθή έκδοση, που σχεδιάστηκε για τους developers, με τις πιο πρόσφατες καινοτομίες και εξελίξεις.

7.3. Ικανότητα ηγεσίας (Leadership) και τρόπος λήψης αποφάσεων στο παζάρι

Ο Raymond προτείνει ότι όλα τα projects Ελεύθερου Λογισμικού θα πρέπει να έχουν έναν "καλοπροαίρετο δικτάτορα", κάποιου είδους ηγέτη, που είναι, συνήθως, ο ιδρυτής του project, για να μπορέσει να οδηγήσει το project και να έχει πάντοτε τον τελευταίο λόγο όταν πρόκειται να ληφθούν αποφάσεις. Οι ικανότητες που θα πρέπει να διαθέτει αυτό το άτομο, περιλαμβάνουν κυρίως να γνωρίζει πως να δίνει κίνητρα και να συντονίζει ένα project, να κατανοεί τους χρήστες και τους συνδημιουργούς, πως να αναζητά την επίτευξη συναίνεσης, και πως να αξιοποιεί και να ενσωματώνει τον καθένα που έχει κάτι να προσφέρει.

Όπως βλέπετε, δεν κάναμε ακόμη αναφορά σε κανένος είδους τεχνική ικανότητα, σαν ένα από τα πιο σημαντικά προσόντα, αν και κάτι τέτοια πράγματα δεν είναι ποτέ περιττά. Καθώς αυξήθηκαν τόσο το μέγεθος των projects όσο και ο αριθμός των developers που εμπλέκονται σε αυτά, αναδύθηκαν νέοι τρόποι οργάνωσης και λήψης αποφάσεων. Το project του πυρήνα Linux, για παράδειγμα, έχει μία ιεραρχική δομή που βασίζεται στον Linus Torvalds, που μοιράζει αρμοδιότητες και είναι ο "καλοπροαίρετος δικτάτορας". Θα δούμε, επίσης, ότι υπάρχουν και επί μέρους κομμάτια του Linux που έχουν τους δικούς τους "καλοπροαίρετους δικτάτορες", αν και η εξουσία τους περιορίζεται από το γεγονός ότι ο Linus Torvalds έχει πάντοτε τον τελευταίο λόγο. Αυτή η περίπτωση αποτελεί ένα σαφές παράδειγμα για το πως μία δομή βασισμένη σε μεγάλο βαθμό σε επί μέρους υποενότητες (modularity) σε ένα project Ελεύθερου Λογισμικού, οδήγησε σε έναν ιδιαίτερο τρόπο οργάνωσης των πραγμάτων και λήψης των αποφάσεων (Alessandro Narduzzo και Alessandro Rossi, "Modularity in action: GNU/Linux and free/ open source software development model unleashed", 2003) [176].

Σημείωση

Μερικοί ισχυρίζονται ότι ο τρόπος οργάνωσης των projects Ελεύθερου Λογισμικού μοιάζει με μία χειρουργική ομάδα,

όπως πρότεινε ο Harlan Mills (of IBM) στις αρχές της δεκαετίας του 1970, άποψη που, κατόπιν, διαδόθηκε και από τον Brooks, με το διάσημο βιβλίο του “The mythical man-month” (Frederick P. Brooks Jr., 1975) [150].

Παρ' όλο που μπορεί να υπάρχουν περιπτώσεις με μία ομάδα ανάπτυξης για μία εφαρμογή ελεύθερου λογισμικού που να περιλαμβάνει έναν designer/developer (τον χειρουργό) και πολλούς συνδημιουργούς (co-developers) που επιτελούν βοηθητικούς ρόλους (διαχειριστές συστημάτων, συντήρηση, εξειδικευμένα καθήκοντα, παραγωγή εγχειριδίων και τεκμηρίωσης), εν τούτοις δεν υφίσταται σχεδόν ποτέ ένας τόσο αυστηρός και σαφώς καθορισμένος διαχωρισμός σαν εκείνον που εισηγήθηκαν οι Mills και Brooks. Στο Ελεύθερο Λογισμικό, όπως επισημαίνει ο Brooks, σύμφωνα με τη λογική της χειρουργικής ομάδας, ο συνολικός αριθμός developers (ή τουλάχιστον των πλέον δραστήριων), που πρέπει να επικοινωνούν για να δημιουργήσουν ένα μεγάλο και σύνθετο σύστημα, είναι πολύ χαμηλότερος σε σχέση με το συνολικό αριθμό των developers.

Στην περίπτωση του Apache Foundation, επικρατεί η *αξιοκρατία*, αφού αυτό το Ίδρυμα προβλέπει μία Διευθυντική Επιτροπή αποτελούμενη από ανθρώπους που συνεισέφεραν σημαντικά στο project. Στην πραγματικότητα, δεν πρόκειται για μία αυστηρή αξιοκρατία, αφού δεν κυβερνούν εκείνοι που συνεισφέρουν τα περισσότερα, με δεδομένο ότι η Διευθυντική Επιτροπή εκλέγεται δημοκρατικά ανά τακτά διαστήματα από τα Μέλη του Ιδρύματος (υπεύθυνο για την διαχείριση διαφόρων projects Ελεύθερου Λογισμικού, όπως το Apache, η Jakarta κλπ). Για να γίνει κανείς μέλος του Ιδρύματος Apache, πρέπει να έχει συνεισφέρει σημαντικά και κατ' εξακολούθηση σε ένα ή περισσότερα από τα projects του Ιδρύματος. Αυτό το σύστημα χρησιμοποιείται και από άλλα μεγάλα projects, σαν το FreeBSD ή το GNOME.

Μία άλλη ενδιαφέρουσα περίπτωση επίσημης οργάνωσης είναι το GCC Steering Committee. Δημιουργήθηκε αρχικά το 1998 για να μη μπορεί οποιοσδήποτε να αναλάβει τον έλεγχο του GCC project (Σύλλογή μεταγλωττιστή του GNU, το σύστημα μεταγλώττισης του GNU) και, λίγους μήνες αργότερα, υποστηρίχθηκε από το Ίδρυμα Ελεύθερου Λογισμικού FSF (που είναι ο βασικός προαγωγός του έργου GNU). Κατά μία έννοια, αυτή η Επιτροπή συνεχίζει την παράδοση μίας άλλης αντίστοιχης επιτροπής του EGCS project (που για κάποιο διάστημα έτρεχε παράλληλα με το GCC project, αλλά αργότερα ενσωματώθηκε σε αυτό). Η βασική του αποστολή είναι να διασφαλίσει ότι το GCC project πληροί *τους στόχους και τα διακηρυγμένα οράματα* που αναφέρονται στο *Καταστατικό* του project. Τα μέλη της Επιτροπής συμμετέχουν ως ιδιώτες και επιλέγονται από το ίδιο το project, με τέτοιο τρόπο που να αντιπροσωπεύουν πιστά τις διάφορες κοινότητες που συνεργάζονται για την ανάπτυξη του GCC (προγραμματιστές υποστήριξης για διάφορες γλώσσες προγραμματισμού, προγραμματιστές που απασχολούνται με τον πυρήνα, ομάδες με ενδιαφέρον για τον προγραμματισμό ενσωματωμένων (embedded) συστημάτων, κλπ.

Ο ηγέτης ενός project Ελεύθερου Λογισμικού δεν απαιτείται να είναι συνεχώς το ίδιο άτομο. Βασικά, υπάρχουν δύο περιστάσεις στις οποίες ένας ηγέτης σε ένα project παύεται. Η πρώτη είναι η απώλεια ενδιαφέροντος, χρόνου ή κινήτρων για να συνεχίσει. Σε αυτή την περίπτωση, το σκήπτρο πρέπει να περάσει σε κάποιον άλλο developer που θα αναλάβει τον ρόλο του project leader. Πρόσφατες μελέτες (Jesús M. González Barahona και Gregorio Robles, 2003) [87] δείχνουν ότι, σε γενικές γραμμές, η ηγεσία του project αλλάζει συχνά χέρια, κατά τέτοιο τρόπο που να μπορούμε, συν τω χρόνω, να δούμε διάφορες γενιές developers. Η δεύτερη περίπτωση είναι πιο προβληματική: περιλαμβάνει ένα forking (κλώνο του αρχικού project). Οι Άδειες του Ελεύθερου Λογισμικού επιτρέπουν να παίρνει κανείς τον κώδικα, να τον τροποποιεί, και να τον αναδιανέμει σε οποιονδήποτε, χωρίς να απαιτείται άδεια από τον ηγέτη του project. Αυτό όμως συνήθως δεν συμβαίνει, εκτός από τις περιπτώσεις εκείνες όπου η ιδέα είναι να αποφευχθεί σκόπιμα ο ηγέτης του project (και το εν δυνάμει βέτο που διαθέτει ενάντια σε μία συνεισφορά). Αυτό, από τη μία μεριά, μοιάζει με ένα είδος “πραξικοπήματος”, το οποίο, από την άλλη μεριά, είναι πλήρως νόμιμο και έγκυρο. Για τον λόγο αυτό, ένας από τους στόχους του ηγέτη για να κρατά ικανοποιημένους

τους συνδημιουργούς (co-developers) είναι να ελαχιστοποιεί την πιθανότητα ενός forking.

7.4. Διαδικασίες του Ελεύθερου Λογισμικού

Αν και το Ελεύθερο Λογισμικό δεν συνδέεται απαραίτητως με κάποια ειδική διαδικασία ανάπτυξης λογισμικού, υφίσταται πάντως μία ευρεία συναίνεση ως προς τις διαδικασίες που συνήθως χρησιμοποιεί. Αυτό, βέβαια, δεν συνεπάγεται ότι δεν υπάρχουν projects ελεύθερου Λογισμικού που να δημιουργήθηκαν χρησιμοποιώντας κλασικές διαδικασίες, όπως το μοντέλο του καταρράκτη. Σε γενικές γραμμές, το αναπτυξιακό μοντέλο των projects Ελεύθερου Λογισμικού είναι περισσότερο ανεπίσημο, διότι οι περισσότεροι από την ομάδα ανάπτυξης εκτελούν αυτές τις εργασίες εθελοντικά και όχι σε ανταλλαγή για κάποια χρηματική αμοιβή, όχι τουλάχιστον με άμεσο τρόπο.

Ο τρόπος με τον οποίο γίνεται ο εντοπισμός των πιθανών αναγκών των χρηστών, στον κόσμο του Ελεύθερου Λογισμικού, εξαρτάται τόσο από την “ηλικία”, όσο και από το μέγεθος του project. Στα αρχικά στάδια, ο ιδρυτής του project και ο χρήστης είναι το ίδιο και το αυτό πρόσωπο. Αργότερα, εφ' όσον το project επεκταθεί, ο εντοπισμός των πιθανών αναγκών των χρηστών τείνει να γίνεται μέσω των λιστών ηλεκτρονικού ταχυδρομείου (mailing lists) και αρχίζει να διαμορφώνεται μία σαφής διάκριση ανάμεσα στην ομάδα ανάπτυξης (ή τουλάχιστον, τους πιο δραστήριους developers) και τους χρήστες. Για μεγάλα projects, με πολυάριθμους χρήστες και πολλούς developers, οι πιθανές ανάγκες των χρηστών εντοπίζονται με τη χρήση του ίδιου εργαλείου σαν και εκείνου που χρησιμοποιείται για τη διαχείριση των σφαλμάτων (bugs).

Σε αυτή την περίπτωση, αντί να μιλάμε για διάφορα σφάλματα (bugs), αναφερόμαστε σε δραστηριότητες μεγαλύτερης κλίμακας, αν και ο μηχανισμός που χρησιμοποιείται για την διαχείρισή τους είναι ταυτόσημος με εκείνον για την αποσφαλμάτωση (θα ταξινομηθούν κατά σειρά σπουδαιότητας, ανάλογα με τις εξαρτήσεις που απαιτούνται, κλπ. και θα είναι εφικτό να παρακολουθεί κανείς κατά πόσο ολοκληρώθηκαν ή όχι). Η χρήση αυτού του εργαλείου οργάνωσης και σχεδίασης είναι σχετικά πρόσφατη. Έτσι, μπορούμε να διαπιστώσουμε πως ο κόσμος του Ελεύθερου Λογισμικού εξελίχθηκε, μέχρι ενός σημείου, από την πλήρη έλλειψη, μέχρι το συγκεντρωτικό σύστημα διαχείρισης αυτών των δραστηριοτήτων, με όρους engineering, έστω και σε περιορισμένο βαθμό. Συνοψίζοντας, δεν είναι συνηθισμένο να βρούμε ένα έγγραφο που να καλύπτει όλες τις προϋποθέσεις, πράγμα που είναι το σύνθημα με το μοντέλο του καταρράκτη.

Όσο αφορά τη συνολική σχεδίαση του συστήματος, μόνο τα μεγάλα projects διαθέτουν κάτι τέτοιο, τεκμηριωμένο εγγράφως, με όλες τις απαραίτητες λεπτομέρειες. Κατά τα λοιπά, ο βασικός developer (είτε η ομάδα των βασικών developers) είναι πολύ πιθανό να τα έχουν όλα αυτά μέσα στο μυαλό τους. Μερικές φορές, ακόμη ούτε κι' αυτό δεν ισχύει, οπότε το σύστημα διαμορφώνεται καθώς εξελίσσεται το πρόγραμμα.

Η απουσία λεπτομερειακού σχεδίου δεν επιβάλλει μόνο περιορισμούς για μία πιθανή επαναχρησιμοποίηση των διαφόρων αρθρωμάτων (*modules*), αλλά συνιστά και ένα σοβαρό εμπόδιο όταν πρόκειται να δώσει κανείς πρόσβαση σε νέους developers, αφού θα βρεθούν αντιμέτωποι με μία αργή και δαπανηρή διαδικασία εκμάθησης. Το να υπάρχει ένα λεπτομερές σχέδιο, δεν είναι ούτε και αυτό συνηθισμένο. Η έλλειψή του σημαίνει ότι πολλές ευκαιρίες για επαναχρησιμοποίηση του κώδικα πάνε χαμένες.

Η φάση της υλοποίησης είναι εκείνη πάνω στην οποία οι περισσότεροι developers Ελεύθερου Λογισμικού επικεντρώνουν τις περισσότερες προσπάθειες, μεταξύ άλλων διότι, κατά την άποψή τους, είναι σαφώς και η πιο διασκεδαστική.

Για να το πετύχουν αυτό, το κλασικό μοντέλο προγραμματισμού που προβλέπει μία διαδικασία δοκιμής και λάθους, τηρείται κανονικά, μέχρι την επίτευξη των επιθυμητών αποτελεσμάτων, τουλάχιστον από την υποκειμενική σκοπιά του προγραμματιστή. Ιστορικά μιλώντας, είναι σπάνιο να περιλαμβάνονται δοκιμές μονάδων λογισμικού (*unit tests*) μαζί με τον κώδικα, έστω κι' αν θα καθιστούσαν τις τυχόν τροποποιήσεις ή ενσωματώσεις περαιτέρω νέου κώδικα εκ μέρους άλλων developers, πολύ πιο εύκολη υπόθεση. Στην περίπτωση κάποιων μεγάλων projects, πχ. σαν το Mozilla, υπάρχουν μηχανήματα ειδικά αφιερωμένα για να κατεβάζουν αποθετήρια που περιέχουν τον πιο πρόσφατο κώδικα και να τον μεταγλωττίζουν σε διαφορετικές αρχιτεκτονικές ("An overview of the software engineering process and tools in the Mozilla project", 2002) [193]. Τα διάφορα bugs που εντοπίζονται αναφέρονται στην mailing list των developers.

Πάντως, οι αυτοματοποιημένες δοκιμές δεν αποτελούν καθιερωμένη πρακτική. Σε γενικές γραμμές, οι ίδιοι οι χρήστες, με την τεράστια ποικιλία χρήσεων που κάνουν, σε διαφορετικές αρχιτεκτονικές και συνδυασμούς, θα εκτελέσουν αυτές τις δοκιμές. Αυτό έχει το πλεονέκτημα ότι τρέχουν πάμπολλες δοκιμασίες παράλληλα και με ελάχιστο κόστος για την ομάδα ανάπτυξης. Το πρόβλημα με αυτό το μοντέλο είναι πως να συλλεχθεί η ανάδραση (*feedback*) από όλους τους χρήστες, με τρόπο που να μπορεί να οργανωθεί και να αξιοποιηθεί όσο καλύτερα γίνεται.

Όσο αφορά την ανάληψη ή όχι του έργου συντήρησης του λογισμικού, μέσα στον κόσμο του Ελεύθερου Λογισμικού, με την έννοια της συντήρησης των προηγούμενων εκδόσεων, αυτό θα εξαρτηθεί από το project. Για projects που χρειάζονται σταθερότητα, όπως οι πυρήνες ενός λειτουργικού συστήματος, οι προηγούμενες εκδόσεις συντηρούνται, διότι η αλλαγή σε μία νέα έκδοση μπορεί να αποβεί τραυματική. Σε γενικές γραμμές, όμως, για τα περισσότερα projects Ελεύθερου Λογισμικού, αν βρεθεί κάποιο bug σε μία προηγούμενη έκδοση, τότε οι developers συνήθως θα το αγνοήσουν και θα συστήσουν τη χρήση της πιο πρόσφατης έκδοσης, με την ελπίδα ότι το bug θα έχει εξαφανισθεί με την εξέλιξη του software.

7.5. Κριτική του βιβλίου “Ο Καθεδρικός ναός και το παζάρι”

Το βιβλίο “Ο Καθεδρικός και το παζάρι”, με δεδομένο το δημοσιογραφικό παρά το επιστημονικό ύφος του κειμένου, πάσχει από έλλειψη συστηματικότητας και αυστηρότητας. Η πιο συχνή κριτική επικεντρώνεται στο γεγονός ότι το βιβλίο, βασικά, εξηγεί την ιδιαίτερη περίπτωση του Linux και προσπαθεί να επεκτείνει τα συμπεράσματα σε όλο τα projects Ελεύθερου Λογισμικού.

Κατ' αυτή την έννοια, στο "Cave or community? An empirical examination of 100 mature open source projects" (“Σπήλαιο ή Κοινότητα; Μία εμπειρική εξέταση 100 ώριμων projects Ανοικτού Κώδικα”) [160] μπορούμε να διαπιστώσουμε ότι η ύπαρξη μίας κοινότητας τόσο μεγάλης σαν και εκείνη που σχηματίστηκε γύρω από τον πυρήνα Linux, είναι μάλλον μία εξαίρεση παρά ο κανόνας. Ακόμη πιο επικριτικοί είναι εκείνοι που πιστεύουν ότι το Linux αποτελεί ένα παράδειγμα του μοντέλου ανάπτυξης τύπου καθεδρικού ναού. Προβάλλουν το επιχείρημα πως, προφανώς, υπάρχει μία κινητήριος δύναμη, ή τουλάχιστον ένα άτομο με το μέγιστο κύρος, καθώς και ένα ιεραρχικό σύστημα που αναθέτει υπευθυνότητες προς την κάτω πλευρά της κλίμακος, τους εργάτες-προγραμματιστές. Υπάρχει, επίσης, μια κατανομή αρμοδιοτήτων, αν και υπονοείται. Το κείμενο "A second look at the cathedral and the bazaar" [91] (“Μία δεύτερη ματιά στον Καθεδρικό και το παζάρι”) πάει ακόμη παραπέρα και υποστηρίζει, όχι χωρίς κάποια στοιχεία πικρίας και αλαζονείας στο σκεπτικό του, ότι η αναλογία με το παζάρι παρουσιάζει μία εσωτερική αντίφαση.

Ένα άλλο από τα σημεία που δέχθηκαν την περισσότερη κριτική από το "Ο Καθεδρικός και το παζάρι" είναι ο ισχυρισμός ότι ο νόμος του Brooks, που λέει ότι "προσθέτοντας developers σε ένα

καθυστερημένο project λογισμικού, το καθυστερεί ακόμη παραπάνω" (*The mythical man-month. Δοκίμια για το software engineering*, 1975) [150], δεν ισχύει στον κόσμο του Ελεύθερου Λογισμικού.

Στην βιβλιογραφική αναφορά [148] μπορούμε να διαβάσουμε πως εκείνο που συμβαίνει στην πραγματικότητα είναι ότι τα περιβαλλοντικά πλαίσια στο ελεύθερο λογισμικό είναι διαφορετικά και πως, επί της αρχής, εκείνο που μοιάζει να είναι ασύμφωνο με τον νόμο του Brooks, μετά από πιο εμβριθή ανάλυση, δεν είναι παρά ένας αντικατοπτρισμός.

7.6. Ποσοτικές μελέτες

Το Ελεύθερο Λογισμικό καθιστά εφικτή την εις βάθος μελέτη του κώδικα, καθώς και όλων εκείνων των παραμέτρων που επηρεάζουν τη δημιουργία του, χάρη στη δυνατότητα πρόσβασης σε πολλές πηγές δημόσιας πληροφόρησης. Αυτό επιτρέπει την υιοθέτηση και ενθάρρυνση πολλών περιοχών της παραδοσιακής Τεχνολογίας Λογισμικού, όπως η εμπειρική Τεχνολογία Λογισμικού, χάρις ακριβώς στην ύπαρξη μίας τεράστιας ποσότητας πληροφορίας που είναι εύκολα διαθέσιμη, χωρίς να χρειάζεται κάποιος να εμβαθύνει πάρα πολύ στην ανάπτυξη του Ελεύθερου Λογισμικού. Οι συγγραφείς είναι πεπεισμένοι ότι αυτή η θεώρηση πραγμάτων μπορεί να συνεισφέρει τα μέγιστα στην ανάλυση και την κατανόηση των φαινομένων που συνδέονται με το Ελεύθερο Λογισμικό (και το λογισμικό, γενικότερα), και ότι θα μπορούσε, μεταξύ άλλων δυνατοτήτων, να καταφέρει να οδηγήσει στην παραγωγή μοντέλων πρόβλεψης για το λογισμικό, με ανάδραση (*feedback*) σε πραγματικό χρόνο.

Η ιδέα που υπάρχει από πίσω είναι πολύ απλή: "αφού έχουμε την δυνατότητα μελέτης ενός τεράστιου αριθμού προγραμμάτων Ελεύθερου Λογισμικού, τότε ας το κάνουμε". Επιπλέον, πέρα και από την παρούσα κατάσταση ενός project, όλη η προηγούμενη εξέλιξή του είναι δημοσίως διαθέσιμη, με την έννοια ότι όλες αυτές οι πληροφορίες, αν συλλεχθούν δεόντως, αναλυθούν και πακεταρισθούν, θα μπορούσαν να χρησιμεύσουν σαν μία βάση γνώσης που θα μας επιτρέψει να αποτιμήσουμε την κατάσταση υγείας ενός project, συνεισφέροντας έτσι στη λήψη αποφάσεων και στην πρόβλεψη των τρεχουσών αλλά και των μελλοντικών επιπλοκών.

Η πρώτη ποσοτική μελέτη κάποιας σημασίας, στον κόσμο του Ελεύθερου Λογισμικού, ανάγεται πίσω στο 1998, αν και είχε δημοσιευθεί στις αρχές του 2000 ("The Orbiten free software survey") [127]. Ο σκοπός της ήταν να εντοπίσει, με εμπειρικό τρόπο, το βαθμό συμμετοχής των developers στο Ελεύθερο Λογισμικό. Για να το πετύχουν, έγινε επεξεργασία των στατιστικών των στοιχείων ταυτοποίησης που προκύπτουν κατά τη συγγραφή, από τις κεφαλίδες (headers) των αρχείων του πηγαίου κώδικα.

Τα αποτελέσματα έδειξαν ότι ο βαθμός συμμετοχής ήταν συμβατός με τον νόμο του Pareto ("Course of Political Economy", Lausane, 1896) [182]: Το 80% του κώδικα αντιστοιχεί στο 20% των πλέον δραστήριων developers, ενώ το υπόλοιπο 80% των developers συνεισφέρουν το 20% του συνολικού κώδικα. Πολλές κατοπινές μελέτες επιβεβαίωσαν και επεξετέιναν την ισχύ αυτού του πρώτου συμπεράσματος σε διαφορετικές μορφές συμμετοχής και συνεισφοράς στον πηγαίο κώδικα (mailing lists, αναφορές των bugs ή ακόμη και αριθμός των downloads, όπως μπορούμε να δούμε στη διεύθυνση:

<http://www.mmd.eng.cam.ac.uk/people/fhh10/Sourceforge/Sourceforge%20paper.pdf> [145]).

Σημείωση

Το γεγονός ότι στις μελέτες για την Τεχνολογία του Ελεύθερου Λογισμικού εμφανίζονται πολλοί οικονομικοί όροι, είναι το αποτέλεσμα του ενδιαφέροντος που έδειξαν μερικοί οικονομολόγοι για να μάθουν και να κατανοήσουν τι

τους εθελοντές να παράγουν αγαθά υψηλής αξίας, χωρίς συνήθως να προσπορίζονται κανένα άμεσο όφελος σε αντάλλαγμα. Η πιο γνωστή δημοσίευση είναι το "Cooking pot markets: an economic model for the trade in free goods and services on the Internet" [125] ("Αγορές της κατσαρόλας: ένα οικονομικό μοντέλο για την εμπορική διακίνηση ελεύθερων αγαθών και υπηρεσιών στο Διαδίκτυο"), που προτείνει την έννοια της Οικονομίας των δώρων στο Διαδίκτυο. Στη διεύθυνση <http://www.wikipedia.org/wiki/Pareto> [232] μπορούμε να βρούμε περισσότερες πληροφορίες για την αρχή του Pareto και τη γενίκευσή της στην κατανομή του Pareto. Η καμπύλη του Lorenz (http://www.wikipedia.org/wiki/Lorenz_curve) [231], που αναπαριστά την συμμετοχή των developers σε ένα project, είναι επίσης πολύ ενδιαφέρουσα, όπως και ο συντελεστής Gini (http://www.wikipedia.org/wiki/Gini_coefficient) [230], που υπολογίστηκε πάνω στη βάση της καμπύλης του Lorenz και ο οποίος παρέχει έναν αριθμό που αναδεικνύει την ανισότητα του συστήματος.

Το εργαλείο που χρησιμοποιήθηκε για τη διεξαγωγή αυτής της μελέτης, δημοσιεύθηκε από τους συγγραφείς της υπό μία Ελεύθερη Άδεια. Συνεπώς, τα αποτελέσματά της μπορούν να αναδημοσιεύονται και να χρησιμοποιούνται για διεξαγωγή νέων μελετών.

Σε μία μεταγενέστερη μελέτη ("Results from software engineering research into open source development projects using public data", 2000) [158] ("Αποτελέσματα έρευνας του software engineering, πάνω σε projects Ανοικτού Κώδικα, με την χρήση δημόσιων δεδομένων"), ο Koch πήγε παραπέρα και ανέλυσε τις αλληλεπιδράσεις σε ένα project Ελεύθερου Λογισμικού. Οι πηγές πληροφορίας ήταν οι mailing lists και το αποθετήριο των διαφόρων εκδόσεων του έργου GNOME. Η πιο ενδιαφέρουσα πλευρά, πάντως, της μελέτης του Koch ήταν η οικονομική ανάλυση. Ο Koch εστιάζεται στον έλεγχο της αξιοπιστίας των κλασικών μεθόδων πρόβλεψης κόστους (Βαθμοί λειτουργίας, Παραμετρικό μοντέλο COCOMO...) και αναδεικνύει τα προβλήματα που συνεπιφέρει η εφαρμογή τους. Ο ίδιος, βέβαια, παραδέχεται ότι τα αποτελέσματα αυτά θα πρέπει να εκλαμβάνονται με τη δέουσα επιφύλαξη, διότι συμπίπτουν μόνον εν μέρει με την πραγματικότητα. Συμπεραίνει ότι το Ελεύθερο Λογισμικό χρειάζεται τα δικά του μοντέλα και τις μεθόδους μελέτης, αφού τα ήδη γνωστά δεν ταιριάζουν στη φύση του. Από την άλλη μεριά, αφού είμαστε, προφανώς, σε θέση να έχουμε τα περισσότερα από τα στοιχεία που σχετίζονται με την ανάπτυξη του Ελεύθερου Λογισμικού σε δημόσια διάθεση, αυτό μας επιτρέπει να είμαστε αισιόδοξοι ως προς την επίτευξη αυτών των στόχων στο εγγύς μέλλον. Η μελέτη του Koch μπορεί να θεωρηθεί ως η πρώτη πλήρως ποσοτική ανάλυση, έστω και αν σίγουρα της λείπει μία σαφής μεθοδολογία, και ακόμη περισσότερο κάποια *ad hoc* εργαλεία που θα επέτρεπαν όχι μόνο την επαλήθευση των αποτελεσμάτων της, αλλά και την διεξαγωγή μελετών άλλων projects.

Το 2000, ο Mockus et al. παρουσίασε την πρώτη μελέτη σε projects Ελεύθερου Λογισμικού που περιελάμβανε μία πλήρη περιγραφή της αναπτυξιακής διαδικασίας και των οργανωτικών δομών, πλαισιωμένη με ποιοτικά και ποσοτικά στοιχεία ("*A case study of open source software development: the Apache server*") [172] ("Μία μελέτη περίπτωσης ανάπτυξης Λογισμικού Ανοικτού Κώδικα: ο Apache server"). Για να το πετύχει, χρησιμοποίησε το αρχείο καταγραφής αλλαγών του προγράμματος και τις αναφορές των bugs, για να ποσοτικοποιήσει την συμμετοχή των developers, το μέγεθος της βασικής ομάδας, την συγγραφική πατρότητα του κώδικα, την παραγωγικότητα, την πυκνότητα σφαλμάτων, και τα διαστήματα που απαιτήθηκαν για την επίλυση των προβλημάτων.

Κατά κάποιο τρόπο, αυτή η μελέτη αποτελεί ακόμη μία κλασική μελέτη Τεχνολογίας Λογισμικού, εκτός από το ότι τα δεδομένα συγκεντρώθηκαν εξ' ολοκλήρου με ημιαυτόματη επισκόπηση των στοιχείων που τα ίδια τα projects παραχωρούν δημοσίως στο Διαδίκτυο. Όπως και στη περίπτωση του "Results from software engineering research into open source development projects using public data", 2000 [158] ("Αποτελέσματα από την έρευνα software engineering σε projects Ανοικτού Κώδικα με χρήση Δημόσιων δεδομένων"), αυτή η δημοσίευση δεν παρείχε κανένα εργαλείο ή μία αυτοματοποιημένη διαδικασία που να μπορεί να χρησιμοποιηθεί στο μέλλον από άλλες ερευνητικές ομάδες.

Στο *"Estimating Linux's size"*, 2000 [227] (*"Εκτιμώντας το μέγεθος του Linux"*), και στο *"More than a gigabuck: estimating GNU/Linux"* [228] (*"Παραπάνω από ένα δισεκατομμύριο δολλάρια: εκτιμώντας το GNU/ Linux"*), βρίσκουμε μία ποσοτική ανάλυση των γραμμών κώδικα και των προγραμματιστικών γλωσσών που χρησιμοποιήθηκαν για την διανομή της Red Hat. Ο González Barahona et al. ακολούθησαν αυτά τα βήματα σε μία σειρά άρθρων σχετικά με την διανομή Debian {βλέπε, πχ. *"Anatomy of two GNU/Linux distributions"* (*"Ανατομία δύο διανομών GNU/Linux"*) [88] }. Όλα αυτά προσφέρουν ένα είδος ακτινογραφίας αυτών των διανομών GNU/Linux, με βάση τα δεδομένα που παρείχε ένα εργαλείο που συμπεριλαμβάνει τις γραμμές πηγαίου κώδικα ενός προγράμματος (SLOC, γραμμές κώδικα που δεν είναι κενές και δεν είναι σχόλια/ comments). Πέρα από το θεαματικό αποτέλεσμα σε συνολικές γραμμές κώδικα (η Debian 3.0, γνωστή και σαν Woody, είχε παραπάνω από 100 εκατομμύρια γραμμές κώδικα), μπορούμε να δούμε πως κατανέμεται ο αριθμός γραμμών κώδικα για κάθε προγραμματιστική γλώσσα. Έχοντας την δυνατότητα να μελετήσουμε την εξέλιξη των διαφορών εκδόσεων του Debian μέσα στον χρόνο, απέφερε κάποια ενδιαφέροντα αποτελέσματα [88]. Αξίζει να σημειωθεί ότι, τα τελευταία 5 χρόνια, το μέσο μέγεθος των πακέτων παρέμεινε πρακτικά σταθερό, που σημαίνει ότι η φυσική τάση για μεγάλωμα, αντισταθμίστηκε με την ενσωμάτωση μικρότερων πακέτων. Συγχρόνως, μπορούμε να διαπιστώσουμε ότι η σημασία της προγραμματιστικής γλώσσας C, αν και ακόμη κυρίαρχη, μειώθηκε με τα χρόνια, ενώ οι γλώσσες script (Python, PHP και Perl) και η Java, γνωρίζουν μία εκρηκτική αύξηση. Οι "κλασικές" μεταγλωττισμένες γλώσσες (Pascal, Ada, Modula...) σχεδόν εξαφανίζονται. Τέλος, αυτά τα άρθρα περιλαμβάνουν ένα κεφάλαιο που δείχνει τα αποτελέσματα που θα έχουμε αν εφαρμόσουμε το κλασικό παραμετρικό μοντέλο COCOMO για μία εκτίμηση κόστους, που προέρχεται από τις αρχές της δεκαετίας του 1980 (*Software Engineering Economics*, 1981) [93] και που χρησιμοποιείται από τα ιδιοταγή λογισμικά για να εκτιμήσουν την προσπάθεια, τα χρονοδιαγράμματα και τα κόστη.

Αν και προκαταρκτικές, οι περισσότερες μελέτες που παρουσιάστηκαν σε αυτό το κεφάλαιο είναι αρκετά περιορισμένες στα projects που έθεσαν υπό εξέταση. Η μεθοδολογία που επιλέχθηκε και προσαρμόστηκε για το υπό εξέταση project, εκτελέστηκε χειροκίνητα και το αυτοματοποιημένο κομμάτι μπορεί, μόνο περιστασιακά, να χρησιμοποιηθεί γενικά και σε άλλα projects Ελεύθερου Λογισμικού. Αυτό σημαίνει ότι η προσπάθεια που απαιτείται για τη μελέτη ενός νέου project θα είναι πολύ μεγαλύτερη, αφού η μέθοδος χρειάζεται να προσαρμοσθεί ξανά και οι χειροκίνητες διαδικασίες θα πρέπει να εκτελεστούν και πάλι.

Για το λόγο αυτό, οι πιο πρόσφατες προσπάθειες (*"Studying the evolution of libre software projects using publicly available data"*, στα: Πρακτικά του 3^{ου} Workshop για το Software Engineering Ανοικτού Κώδικα, 25ο Διεθνές Συνέδριο για το software engineering, Portland, ΗΠΑ [196] ή *"Automating the measurement of open source projects"*, 2003 [124]) εστιάζονται στη δημιουργία μίας υποδομής ανάλυσης που να ενσωματώνει διάφορα εργαλεία με τρόπο ώστε η διαδικασία να μπορεί να αυτοματοποιηθεί στο έπακρο.

Υπάρχουν δύο αρκετά προφανείς λόγοι για να γίνει αυτό: ο πρώτος είναι ότι άπαξ και επενδύθηκε πολύς χρόνος και προσπάθεια για τη δημιουργία ενός εργαλείου ανάλυσης του project, με ιδιαίτερη έμφαση να καταστεί γενικής χρήσης, τότε η παραπάνω προσπάθεια για να χρησιμοποιηθεί και σε άλλα projects ελεύθερου λογισμικού είναι ελάχιστη. Ο δεύτερος είναι ότι η προσπάθεια ανάλυσης, με τη χρήση μίας σειράς εργαλείων που μελετούν τα προγράμματα από διαφορετικές και συχνά συμπληρωματικές σκοπίες, δεν μας επιτρέπει ενίοτε να έχουμε μία γενικότερη αντίληψη του project. Στον ιστότοπο Libre Software Engineering [86] μπορούμε να ακολουθήσουμε αυτές τις πρωτοβουλίες με μεγαλύτερη λεπτομέρεια.

7.7. Μελλοντική εργασία

Μετά την περιγραφή της σύντομης αλλά έντονης ιστορίας της έρευνας της τεχνολογίας λογισμικού στο Ελεύθερο Λογισμικό, μπορούμε να πούμε πως βρίσκεται ακόμη στα πρώτα της βήματα. Χρειάζεται ακόμη παραπάνω λεπτομερειακή έρευνα και ανάλυση πολλών σημαντικών θεμάτων, μέχρι να βρούμε ένα μοντέλο που να εξηγεί, τουλάχιστον εν μέρει, πως παράγεται το Ελεύθερο Λογισμικό. Τα θέματα που πρέπει να αντιμετωπισθούν στο εγγύς μέλλον, περιλαμβάνουν την κατάταξη των projects ελεύθερου λογισμικού, τη δημιουργία μίας μεθοδολογίας βασισμένης όσο γίνεται περισσότερο σε αυτοματοποιημένη ανάλυση και χρήση κεκτημένης γνώσης για τη δημιουργία μοντέλων που να μας βοηθούν να κατανοήσουμε πως αναπτύσσεται το Ελεύθερο Λογισμικό, αλλά συγχρόνως να μας διευκολύνουν στη λήψη αποφάσεων, με βάση την υπάρχουσα εμπειρία.

Μία άλλη πλευρά που δεν πρέπει να παραβλέπεται και που αρχίζει, ήδη, να προσελκύει την προσοχή, είναι η εγκυρότητα των κλασικών μεθόδων τεχνολογίας στο πεδίο του Ελεύθερου Λογισμικού, σε όλο το φάσμα εντατικοποίησης της τεχνολογίας λογισμικού. Για παράδειγμα, οι νόμοι της Εξέλιξης του Λογισμικού, που πρότεινε ο Lehman (*"Metrics and laws of software evolution - the nineties view"* [165]), στις αρχές της δεκαετίας του 1970 και μετά την επικαιροποίηση και επέκτασή τους στη δεκαετία του 1980 και του 1990, φαίνεται να μην ισχύουν άνευ όρων στην εξέλιξη μερικών projects Ελεύθερου Λογισμικού (*"Understanding open source software evolution: applying, breaking and rethinking the laws of software evolution"*, 2003 [199]).

Στην παρούσα φάση, μία από τις σοβαρότερες ανεπάρκειες είναι η έλλειψη μίας αυστηρής κατηγοριοποίησης, για να μπορούν τα projects Ελεύθερου Λογισμικού να κατατάσσονται σε διάφορες ομάδες.

Τα μέχρι τώρα κριτήρια κατάταξης είναι τόσο ευρέα, που, τελικά, διάφορα projects, με πολύ ανόμοια οργανωτικά, τεχνικά, ή άλλα χαρακτηριστικά, μπαίνουν όλα στο ίδιο σακί. Το επιχείρημα ότι το Linux, με μία μεγάλη κοινότητα και μεγάλο αριθμό developers, έχει μία διαφορετική φύση και δε συμπεριφέρεται όπως ένα πολύ πιο περιορισμένο σε αριθμούς developers και χρηστών project, είναι πολύ αληθινό. Συνολικά, μία λεπτομερέστερη κατάταξη θα καθιστούσε εφικτή την επαναχρησιμοποίηση της εμπειρίας που κερδήθηκε σε άλλα παρόμοια projects (δηλ., με όμοια χαρακτηριστικά), κάνοντας ευκολότερη τη διατύπωση προβλέψεων, πρόγνωση κινδύνων, κλπ.

Το δεύτερο μεγάλο θέμα που πρέπει να αντιμετωπίσει η τεχνολογία του Ελεύθερου Λογισμικού και που σχετίζεται στενά με το προηγούμενο σημείο, αλλά και με τις τρέχουσες τάσεις, είναι η δημιουργία μίας μεθοδολογίας και των εργαλείων για την υποστήριξή της. Μία σαφής και συνοπτική μεθοδολογία θα διευκολύνει τη μελέτη όλων των projects "έπ' ίσοις όροις", την ανακάλυψη της πραγματικής τους κατάστασης, την κατανόηση του πως εξελίχθηκαν και, βέβαια, την κατάταξή τους. Όταν ασχοληθεί κανείς με αυτό το θέμα, τότε τα εργαλεία είναι βασικής σημασίας, επειδή όταν, πλέον, θα έχουν δημιουργηθεί, θα επιτρέπουν την ανάλυση χιλιάδων άλλων projects, με ελάχιστη παραπάνω προσπάθεια.

Ένας από τους σκοπούς του engineering του Ελεύθερου Λογισμικού είναι να κάνει εφικτή τη μελέτη ενός project σε βάθος, στη βάση ενός περιορισμένου συνόλου παραμέτρων, δείχνοντας που μπορεί να εντοπισθούν οι πληροφορίες στο Διαδίκτυο (τη διεύθυνση των αποθετηρίων των διαφόρων εκδόσεων, το μέρος όπου είναι αποθηκευμένα τα αρχεία της mailing list, την τοποθεσία όπου βρίσκεται το σύστημα διαχείρισης των σφαλμάτων και μία στοιχειώδης ανασκόπηση του έργου).

Οι διαχειριστές του project θα έχουν, τότε, με το πατήμα ενός κουμπιού, μία πλήρη ανάλυση, ένα είδος ιατρικού ιστορικού, που θα τους βοηθούσε να διαγνώσουν την κατάσταση υγείας του project, περιλαμβάνοντας ταυτόχρονα και τις περιοχές που επιδέχονται περαιτέρω βελτίωση.

Όταν θα έχουμε αποκτήσει τις μεθόδους, μία κατάταξη και μοντέλα, τότε οι ευκαιρίες που θα μπορούσαν να προκύψουν από την προσομοίωση και, για να είμαστε πιο ακριβείς, αναφερόμαστε στη χρήση ευφύων πρακτόρων (*intelligent agents*), θα ήταν τεράστιες. Λαμβάνοντας υπ' όψη ότι το δικό μας σημείο εκκίνησης είναι ένα εκ προοιμίου σύνθετο σύστημα, θα είχε ενδιαφέρον να δημιουργήσουμε δυναμικά μοντέλα, μέσα στα οποία, οι διάφορες οντότητες που συμμετέχουν στην δημιουργία λογισμικού, να μπορούν να μοντελοποιηθούν. Προφανώς, όσο πιο πολλά γνωρίζουμε για τα διάφορα στοιχεία, τόσο περισσότερο το μοντέλο μας θα είναι προσαρμοσμένο στην πραγματικότητα. Αν και ξέρουμε ότι έχουν προταθεί πολλά και διάφορα μοντέλα προσομοίωσης ελεύθερου λογισμικού, όλα τους είναι αρκετά απλά και ατελή. Σε κάποιο βαθμό, αυτό οφείλεται στο γεγονός ότι υπάρχει ακόμη τεράστια έλλειψη γνώσης σχετικά με τις αλληλεπιδράσεις που λαμβάνουν χώρα κατά τη διαδικασία παραγωγής Ελεύθερου Λογισμικού. Αν μπορούσαμε να οργανώσουμε σωστά και να επεξεργασθούμε τις πληροφορίες για τα διάφορα projects, μέσα από την ιστορία τους, τότε οι έξυπνοι πράκτορες θα μπορούσαν να αποδειχθούν κρίσιμης σημασίας, για να προβλέψουν την κατεύθυνση προς την οποία θα κινηθεί η μελλοντική τους ανάπτυξη. Αν και υπάρχουν πολλές προτάσεις για το πώς να προσεγγίσουμε αυτό το πρόβλημα, μπορούμε να βρούμε μία από τις πιο προχωρημένες στη διεύθυνση <http://www.wai.wu.wien.ac.at/~koch/oss-book/> [82].

7.8. Περίληψη

Συμπερασματικά, προσπαθήσαμε να δείξουμε σε αυτό το κεφάλαιο ότι η Τεχνολογία του Ελεύθερου Λογισμικού αποτελεί ακόμη ένα νέο και ανεξερευνητό πεδίο. Τα πρώτα βήματα που έγιναν οφείλονται σε δημοσιογραφικής φύσης δοκίμια, τα οποία, αν και όχι χωρίς να λείπει μία κάποια επιστημονική αυστηρότητα, πρότειναν την υιοθέτηση ενός πιο αποτελεσματικού μοντέλου. Στο εντωμεταξύ, όμως, έλαβε χώρα μία σταδιακή πρόοδος προς την κατεύθυνση της συστηματικής μελέτης του Ελεύθερου Λογισμικού, από τη σκοπιά του software engineering. Στις μέρες μας, μετά από αρκετά χρόνια αναφορών, ποσοτικών αλλά και ποιοτικών αναλύσεων των ελεύθερων projects, μπορούμε να πούμε ότι κατεβλήθη μία τεράστια προσπάθεια για την επίτευξη μίας συνολικής υποδομής, που καθιστά εφικτή την κατάταξη, ανάλυση και μοντελοποίηση ενός project, πάντοτε εντός ορισμένου χρονικού πλαισίου και κατά τρόπο εν μέρει αυτοματοποιημένο. Όταν η ανάλυση των projects Ελεύθερου Λογισμικού σταματήσει να είναι τόσο δαπανηρή, σε χρόνο, αλλά και σε προσπάθεια, όπως ίσχυε μέχρι τώρα, τότε είναι πιθανό να αρχίσει μία νέα εποχή στο software engineering. Θα εμφανιστούν επί σκηνής διαφορετικοί τύποι τεχνικής, σχεδιασμένοι κυρίως για να προβλέπουν την εξέλιξη του λογισμικού καθώς και τις πιθανές εν δυνάμει επιπλοκές.

8. Περιβάλλοντα ανάπτυξης και Τεχνολογίες

"Τα εργαλεία που χρησιμοποιούμε ασκούν μία βαθιά (και ύπουλη!) επίδραση πάνω στον τρόπο σκέψης μας, και, άρα, πάνω και στην ικανότητά μας για σκέψη".

Edsger W. Dijkstra, "How do we tell truths that might hurt?" (Πώς να πούμε τις αλήθειες που πονάνε;)

Με την πάροδο του χρόνου, τα projects Ελεύθερου Λογισμικού δημιούργησαν τα δικά τους (επίσης ελεύθερα) εργαλεία και συστήματα, για να συνεισφέρουν στη διαδικασία ανάπτυξης. Αν και το κάθε μεμονωμένο project ακολουθεί τους δικούς του κανόνες και χρησιμοποιεί τα δικά του είδη εργαλείων, υπάρχουν ωστόσο ορισμένες πρακτικές, περιβάλλοντα και τεχνολογίες που μπορούν να θεωρηθούν σαν συνηθισμένα στον κόσμο του αναπτυσσόμενου Ελεύθερου Λογισμικού. Σε αυτό το κεφάλαιο θα εξετάσουμε τα πιο κοινά από αυτά και θα συζητήσουμε την επίδρασή τους στη διαχείριση και την ανάπτυξη του project.

8.1. Περιγραφή περιβαλλόντων, εργαλείων και συστημάτων

Πριν προχωρήσουμε στην εξήγηση των ειδικών εργαλείων, πρέπει να ορίσουμε τα γενικά χαρακτηριστικά και τις ιδιότητες, αναλόγως της εργασίας που πρέπει να εκτελεσθεί και του τρόπου οργάνωσης των developers.

Κατ' αρχήν, αν και δεν είναι καθοριστικός παράγοντας, είναι συνηθισμένο τόσο για το περιβάλλον, όσο και για τα εργαλεία ανάπτυξης (ακόμη και για την εικονική μηχανή-στόχο, όπου και όποτε υπάρχει τέτοια), να είναι επίσης ελεύθερα. Αυτό δεν ήταν πάντα έτσι. Για παράδειγμα, το έργο GNU, με στόχο να αντικαταστήσει το Unix, για να αναπτυχθεί μέσα σε και για ιδιοταγή συστήματα Unix, μέχρι την εμφάνιση του Linux και της οικογένειας του BSD. Στις μέρες μας, ειδικά όταν αναπτύσσεται Ελεύθερο Λογισμικό στα πλαίσια ενός επιχειρηματικού μοντέλου, η τάση είναι ότι το μηχανήμα στόχος να μπορεί να τρέχει και ένα ιδιοταγές σύστημα, συχνά με τη μεσολάβηση ενδιάμεσων εικονικών μηχανών (Java, Python, PHP, κλπ.). Σε κάθε περίπτωση, τόσο το ίδιο το περιβάλλον, όσο και η εικονική μηχανή, πρέπει να είναι αρκετά κοινά και φθηνά, για να συγκεντρώσουν μαζί τους συνδημιουργούς (*co-developers*) που διαθέτουν τα ίδια εργαλεία.

Κατά δεύτερο λόγο, ακόμη και για να μπορέσει να προσελκύσουν τον μέγιστο δυνατό αριθμό συνδημιουργών (*co-developers*), τα εργαλεία πρέπει να είναι απλά, σαφώς γνωστά και να μπορούν να λειτουργούν σε οικονομικά μηχανήματα. Ίσως, γι' αυτούς ακριβώς τους λόγους, ο κόσμος του Ελεύθερου Λογισμικού να είναι αρκετά συντηρητικός όταν πρόκειται για γλώσσες, εργαλεία και περιβάλλοντα.

Ως τρίτο σημείο, το μοντέλο ανάπτυξης Ελεύθερου Λογισμικού κλίνει προς το να είναι εξόχως κατανοητό, με πολυάριθμους εν δυνάμει συνεργάτες, μοιρασμένους παντού ανά τον κόσμο. Για τον λόγο αυτό, καθίστανται σε γενικές γραμμές απαραίτητα τα εργαλεία ασύγχρονης συνεργασίας, που ταυτοχρόνως επιτρέπουν στη διαδικασία ανάπτυξης να προχωρήσει εύκολα, ασχέτως του όγκου δουλειάς και του ρυθμού του κάθε συνεργάτη, χωρίς να καθυστερεί κανέναν.

Τέλος, συνιστάται να παρέχονται στους developers διάφορες αρχιτεκτονικές, πάνω στις οποίες να μπορούν να μεταγλωττίζουν και να δοκιμάζουν τα προγράμματα που φτιάχνουν.

8.2. Σχετικές γλώσσες και εργαλεία

Κατά πλειοψηφία, το Ελεύθερο Λογισμικό είναι γραμμένο σε γλώσσα C, όχι μόνο διότι πρόκειται για τη φυσική γλώσσα οποιασδήποτε ποικιλίας Unix-οειδούς συστήματος (που είναι η συνήθης πλατφόρμα του Ελεύθερου Λογισμικού), αλλά και διότι είναι ευρέως διαδεδομένη, τόσο στο μυαλό των ανθρώπων, όσο και στα μηχανήματα (ο GCC είναι ο τυπικός μεταγλωττιστής που είναι εγκατεστημένος από προεπιλογή, σχεδόν σε όλες τις διανομές). Για την ακρίβεια, ακριβώς για τους παραπάνω λόγους αλλά και λόγω της αποτελεσματικότητάς του, ο Stallman συνιστά τη χρήση του στα διάφορα έργα GNU ("*GNU coding standards*") [203]. Άλλες σχετικές όμοιες γλώσσες είναι η C++, που επίσης υποστηρίζεται από προεπιλογή από τον GCC, η Java, που επίσης παρουσιάζει κάποιες ομοιότητες και είναι δημοφιλής διότι επιτρέπει ανάπτυξη λογισμικού για εικονικές μηχανές διαθέσιμες σε μία μεγάλη γκάμα από πλατφόρμες. Σε γενικές γραμμές, δε λαμβάνονται υπ' όψη τα διάφορα σκεπτικά που συναντάμε στην Τεχνολογία Λογισμικού: στο SourceForge (βλέπε κεφάλαιο 8.9.1), το 2004, για κάθε 160 projects σε C, αντιστοιχούσε ένα μόνο σε Ada, παρότι η τελευταία είναι, υποτίθεται, μία πιο κατάλληλη γλώσσα για την ανάπτυξη ποιοτικών προγραμμάτων. Την ίδια στιγμή, τα Αγγλικά καθιερώθηκαν ως η *lingua franca* των developers Ελεύθερου Λογισμικού,

παρά το γεγονός ότι η Esperanto είναι μία πολύ ευκολότερη γλώσσα προς εκμάθηση και με πολύ λογικότερη δομή. Οι Διερμηνευόμενες (*Interpreted*) γλώσσες που σχεδιάστηκαν για την ταχεία κατασκευή πρωτοτύπων δοκιμαστικών εκδόσεων, την προτυποποίηση (*prototyping*) των κανονικών εφαρμογών και των υπηρεσιών web, όπως η Perl, η Python και η PHP, είναι επίσης δημοφιλείς.

Κατά τον ίδιο τρόπο που η C αποτελεί την καθιερωμένη γλώσσα, το *make* είναι το καθιερωμένο εργαλείο για το χτίσιμο προγραμμάτων, λαμβάνοντας υπ' όψη τα αρχεία πηγαίου κώδικα που διαθέτει. Ένας ελεύθερος προγραμματιστής θα χρησιμοποιήσει κανονικά την έκδοση GNU (GNU make) [36] παρά την αντίστοιχη, αλλά ασύμβατη, έκδοση της BSD (Adam de Boor, "*PMake - a tutorial*") [100]. Μπορούν να χρησιμοποιηθούν για την εμφάνιση/ διευκρίνιση των δένδρων με τις εξαρτήσεις μεταξύ των αρχείων, αλλά και των κανόνων για δημιουργία εξαρτημένων αρχείων, από εκείνα από τα οποία εξαρτώνται. Συνεπώς, μπορούμε να διευκρινίσουμε ότι ένα αρχείο με ένα αντικείμενο `x.o`, εξαρτάται από τα αρχεία πηγαίου κώδικα `x.c` και `x.h`, και ότι για να το δημιουργήσουμε, πρέπει να εκτελέσουμε το `gcc -c x.c`. Είτε, ότι τα εκτελέσιμα των προγραμμάτων μας εξαρτώνται από μία συλλογή αντικειμένων και είναι συνδεδεμένα κατά έναν ορισμένο τρόπο. Όταν επιφέρουμε τροποποιήσεις στον πηγαίο κώδικα και μετά εκτελούμε "*make*", μόνο οι εμπλεκόμενες ενότητες (*modules*) θα μεταγλωττιστούν ξανά και το τελικό αντικείμενο θα συνδεθεί εκ νέου.

Αυτό είναι ένα εργαλείο πολύ χαμηλού επιπέδου, αφού, για παράδειγμα, δεν μπορεί να βρει από μόνο του πότε μία ενότητα (*module*) χρειάζεται να μεταγλωττισθεί σε C, παρά το γεγονός ότι θα μπορούσε να το πετύχει εξετάζοντας τις αλληλουχίες των εντολών. Είναι, επίσης, πολύ ισχυρό, αφού μπορεί να συνδυάσει όλα τα διαθέσιμα εργαλεία μετατροπής αρχείων, για να χτίσει ιδιαίτερα σύνθετους στόχους για ένα πολυγλωσσικό project. Αλλά είναι πολύ περίπλοκο και πολύ εξαρτώμενο από Unix-οειδή περιβάλλοντα. Άλλες -υποτίθεται- καλύτερες εναλλακτικές, σαν τη *jam* (*Jam Product Information*) [41], την *aap* (*Aap Project*) [1] ή την *ant* (*The Apache Ant Project*) [7] χρησιμοποιούνται σπάνια (η τελευταία κερδίζει δημοτικότητα κυρίως στον κόσμο της Java).

Με δεδομένη την ετερογένεια των υπαρχόντων συστημάτων, ακόμη και στον κόσμο του Unix, χρησιμοποιούμε και εργαλεία σχεδιασμένα για να μας βοηθήσουν να πετύχουμε φορητότητα για τα προγράμματά μας.

Τα εργαλεία του έργου GNU: "*autoconf*" (<http://www.gnu.org/software/autoconf>) [10], "*automake*" (<http://www.gnu.org/software/automake>) [32] και το "*libtool*" (<http://www.gnu.org/software/libtool>) [35], κάνουν αυτή τη δουλειά ευκολότερη σε γλώσσα C και σε περιβάλλοντα Unix. Λαμβάνοντας υπ' όψη την ετερογένεια των γλωσσών, των ομάδων χαρακτήρων (*character sets*) και των διαφόρων πολιτισμικών υποβάθρων, οι προγραμματιστές σε γλώσσα C (καθώς και εκείνοι που χρησιμοποιούν πολλές άλλες γλώσσες), συχνά χρησιμοποιούν το "*gettext*" (<http://www.gnu.org/software/gettext>) [31] και τις επιλογές διεθνοποίησης (*internationalisation*) που προσφέρει η standard βιβλιοθήκη C (<http://www.gnu.org/software/libc>) [34], για τις προγραμματιστικές εφαρμογές που μπορούν εύκολα να προσαρμοσθούν σε οποιοδήποτε πολιτισμικό υπόβαθρο κατά τη διάρκεια εκτέλεσης του προγράμματος.

Συνεπώς, όταν παραλαμβάνουμε ένα πακέτο πηγαίου κώδικα, θα είναι γραμμένο κατά πάσα πιθανότητα σε γλώσσα C, πακεταρισμένο με το *tar*, συμπιεσμένο με *gzip*, και με δυνατότητες φορητότητας με τη χρήση του *autoconf* και των συναφών εργαλείων, και τέλος, μπορεί να χτισθεί και να εγκατασταθεί με το *make*. Η εγκατάστασή του θα εκτελείται με μία διαδικασία πολύ παρόμοια με εκείνη που ακολουθεί πιο κάτω:

```
tar xzvf package-1.3.5.tar.gz  
cd package-1.3.5  
./configure  
make make install
```

8.3. Ολοκληρωμένα Περιβάλλοντα Ανάπτυξης (*Integrated Development Environments, IDE*)

Ένα Ολοκληρωμένο Περιβάλλον Ανάπτυξης (*Integrated Development Environment, IDE*), είναι ένα σύστημα που κάνει τη δουλειά του προγραμματιστή λογισμικού πιο εύκολη, ολοκληρώνοντας με ισχυρό τρόπο την έκδοση του πηγαίου κώδικα σε κάποια γλώσσα προγραμματισμού, τη μεταγλώττιση ή τη διερμηνεία, την αποσφαλμάτωση, τις μετρήσεις επίδοσης, την ενσωμάτωση πηγαίου κώδικα σε ένα σύστημα ελέγχου πηγαίου κώδικα, κλπ., με έναν συνήθως αρθρωτό (*modular*) τρόπο. Αυτά τα εργαλεία δεν αρέσουν σε όλους τους developers, αν και η χρήση τους, σταδιακά, γενικεύθηκε. Το πρώτο στον κόσμο του Ελεύθερου Λογισμικού που γνώρισε ευρύτατη χρήση ήταν το GNU Emacs (<http://www.gnu.org/software/emacs/>) [33], έργο αιχμής του Richard Stallman, που γράφτηκε και επεκτάθηκε σε Emacs Lisp, για το οποίο υπάρχουν πολλές συνεισφορές.

Το Eclipse (*Eclipse - An Open Development Platform*) [23] μπορεί να θεωρείται το σημερινό IDE αναφοράς στον κόσμο του Ελεύθερου Λογισμικού, με το μειονέκτημα ότι δουλεύει καλύτερα (περί τον Μάιο του 2007) σε μία μη-ελεύθερη εικονική μηχανή Java (εκείνη της Sun, που ελπίζουμε, εν πάσει περιπτώσει, να γίνει σύντομα ελεύθερη). Άλλα δημοφιλή περιβάλλοντα είναι το Kdevelop (<http://www.kdevelop.org>) [42] για το KDE, το Anjuta (<http://www.anjuta.org>) [6] για το GNOME, το Netbeans (<http://www.netbeans.org>) [51] της Sun για την Java και το Code::Blocks (<http://www.codeblocks.org>) [18] για τις εφαρμογές C++.

8.4. Βασικοί μηχανισμοί συνεργασίας

Το Ελεύθερο Λογισμικό είναι ένα φαινόμενο που κατέστη εφικτό χάρη στη συνεργασία διαφόρων καταναμημένων κοινοτήτων και το οποίο, κατά συνέπεια, χρειάζεται τα κατάλληλα εργαλεία για να κάνει αποτελεσματική αυτή την συνεργασία. Παρά το ότι επί μεγάλο χρονικό διάστημα ελάμβανε χώρα η απλή ταχυδρομική αποστολή μαγνητικών ταινιών, η γρήγορη ανάπτυξη του Ελεύθερου Λογισμικού άρχισε ακριβώς όταν έγινε εφικτή η ταχεία επικοινωνία με πολλούς ανθρώπους και, άρα, διευκολύνθηκε τόσο η διανομή του πηγαίου κώδικα προς αυτούς, όσο και η απάντηση με σχόλια και διορθωτικά patches. Χάριν ευκολίας, αντί της αποστολής κώδικα, κατέστη δυνατή η αποστολή μηνυμάτων με διάφορες πληροφορίες προς το site από το οποίο θα μπορούσε πλέον κανείς να κατεβάσει τον κώδικα. Πράγματι, ακριβώς κατά τις αρχές της δεκαετίας του 1970, το e-mail αποτελούσε μία επέκταση του πρωτοκόλλου μεταφοράς αρχείων (FTP) του ARPANET.

Στον κόσμο του Unix, στα μισά της δεκαετίας του 1970, αναπτύχθηκε το *uucp*, το πρωτόκολλο μεταφοράς αρχείων του Unix, για να μπορούν τα μηχανήματα να επικοινωνούν μεταξύ τους μέσω συνδέσεων dial-up και αφιερωμένων γραμμών, και πάνω στο οποίο χτίστηκε το ηλεκτρονικό ταχυδρομείο. Το 1979, εμφανίσθηκε το πρώτο “*USENET link over UUCP. USENET news*”, ένα

ιεραρχικά δομημένο σύστημα forum, που διένειμε μέσω μαζικής αποστολής (*flooding*) προς ιεραρχικά δομημένα sites και που έπαιξε βασικό ρόλο στην ανάπτυξη του Ελεύθερου Λογισμικού, στέλνοντας τον Πηγαίο Κώδικα ολόκληρων προγραμμάτων, προς τις ομάδες `comp.sources`.

Συγχρόνως, αναπτύχθηκαν και *mailing lists*, στα πλαίσια των οποίων οι διαχειριστές της λίστας BITNET (1981) αξίζουν μία ιδιαίτερη μνεία. Σήμερα, βέβαια, προτιμώνται συνήθως οι mailing lists αντί των USENET-type newsgroups. Αυτό συμβαίνει κυρίως λόγω των καταχρήσεων για εμπορικούς σκοπούς και την παρεμβολή "απρόσεκτων" ανθρώπων, που ενοχλούσαν, προσθέτοντας μόνο θόρυβο στις συζητήσεις. Εξ' άλλου, οι mailing lists παρέχουν μεγαλύτερη δυνατότητα ελέγχου και μπορούν να φθάσουν σε πολύ κόσμο. Οι αποδέκτες πρέπει πρώτα να εγγραφούν και οποιαδήποτε ηλεκτρονική διεύθυνση είναι έγκυρη, ακόμη και αν δεν υπάρχει άμεση προσβαση στο Διαδίκτυο. Ο διαχειριστής της mailing list είναι σε θέση να γνωρίζει τόσο το ποιος εγγράφεται, όσο και το ποιος διαγράφεται. Τα διάφορα μηνύματα που συνεισφέρει ο καθένας μπορούν να περιορισθούν μόνο στα μέλη που είναι εγγεγραμμένα ή ο προγραμματιστής μπορεί να επιλέξει να επιβλέπει και να εγκρίνει (*moderate*) τα άρθρα πριν την δημοσίευσή⁶.

⁽⁶⁾Υπάρχουν επίσης και επιβλεπόμενα (*moderated*) news-groups.

Παραδοσιακά, η διαχείριση των *mailing lists* γινόταν μέσω *e-mail*, χρησιμοποιώντας ειδικά μηνύματα με έναν κωδικό πρόσβασης, επιτρέποντας στον διαχειριστή να μην έχει συνεχή πρόσβαση στο Διαδίκτυο, αν και αυτό γίνεται ένα όλο και περισσότερο σπάνιο φαινόμενο. Αυτό σημαίνει ότι η διαχείριση των πιο δημοφιλών mailing lists, στις μέρες μας (*Mailman, the GNU Mailing List Manager*) [46], δεν μπορεί να γίνεται μέσω *e-mail*, αλλά υποχρεωτικά μέσω του Διαδικτύου. Οι mailing lists παίζουν καίριο ρόλο στον κόσμο του Ελεύθερου Λογισμικού και σε πολλές περιπτώσεις⁷ μπορεί να είναι και ο μοναδικός τρόπος για να γίνουν οι συνεισφορές.

Στη σημερινή εποχή, με τη δημοτικότητα του Διαδικτύου, πολλά forums είναι αμιγή web forums ή *weblogs*, δηλαδή χωρίς καμία άλλη διεπαφή εκτός από εκείνη που παρέχει ο περιηγητής διαδικτύου (*navigator*). Μπορεί να είναι γενικής φύσεως, σαν το δημοφιλές SlashDot ("*Slashdot: News for Nerds*") [58] ή το Ισπανικό Barrapunto (<http://barrapunto.com>) [11], όπου αναγγέλλονται και συζητούνται τα καινούργια Ελεύθερα Λογισμικά. Μπορεί, επίσης, να εξειδικεύονται σε ένα ιδιαίτερο πρόγραμμα. Σε αυτή την περίπτωση βρίσκονται συνήθως ενσωματωμένα με διάφορα άλλα πρόσθετα εργαλεία, σε ιστοσελίδες συνεργασίας (βλέπε Κεφάλαιο 8.6.2). Υπάρχουν επίσης και διεπαφές Web για newsgroups καθώς και για παραδοσιακές λίστες.

Ένας άλλος μηχανισμός συνεργασίας που έγινε δημοφιλής κατά την ίδια περίοδο, και που βασίζεται στα wikis, ιδιαίτερα όταν πρόκειται να συνταχθεί ένα έγγραφο κοινής αποδοχής, σαν τις τεχνικές προδιαγραφές ενός προγράμματος, ενός αρθρώματος (*module*) ή ενός συστήματος. Θα τα συζητήσουμε όλα αυτά στο Κεφάλαιο 8.6.2.

⁽⁷⁾ Για παράδειγμα, οι συνεισφορές προς το Linux πρέπει να γίνονται υπό μορφή αρχείων κειμένου (*text patches*) προς τη λίστα linux-kernel@vger.kernel.org.

Τέλος, θα πρέπει να αναφέρουμε τους μηχανισμούς αλληλεπίδρασης που χρησιμοποιούνται από τους developers για να συνδιαλέγονται σε πραγματικό χρόνο. Αυτό δεν αποδείχθηκε και πολύ πρακτικό στην περίπτωση του Ελεύθερου Λογισμικού, καθώς λόγω της μεγάλης διασποράς των developers ανά την υφήλιο, δεν είναι και πολύ εύκολο να βρεθεί μία κοινή ώρα που να τους βολεύει όλους. Παρ' όλα ταύτα, υπάρχουν αρκετά projects που χρησιμοποιούν αυτά τα εργαλεία ανταλλαγής μηνυμάτων κειμένου (*text chat*), είτε σε συστηματική τακτική βάση, είτε κατά τη διάρκεια εικονικών συσκέψεων (*virtual conferences*), σε προσυμφωνημένες ημερομηνίες.

Το εργαλείο που χρησιμοποιείται πιο πολύ, είναι το IRC (*Internet Relay Chat*,

<http://www.ietf.org/rfc/rfc2810.txt>) [151], που θέτει σε επικοινωνία τους ανθρώπους που είναι μέσα σε κάποια "κανάλια", που φτιάχνονται με βάση μία σειρά συνεργαζόμενων διακομιστών. Δεν είναι συχνή η χρήση πολυμεσικών (*multimedia*) εργαλείων (εικόνας, ήχου), πιθανώς επειδή απαιτούνται συνδέσεις υψηλής ποιότητας, που δεν μπορεί να διαθέτει ο καθένας και που, ίσως, επιφέρουν την εμφάνιση προβλημάτων με το διαθέσιμο Ελεύθερο Λογισμικό, καθώς και μία δυσκολία καταγραφής και επεξεργασίας των αποτελεσμάτων των συνομιλιών για λόγους τεκμηρίωσης.

8.5. Διαχείριση πηγαιού κώδικα

Κάθε project, οποιουδήποτε προγράμματος, πρέπει να αρχειοθετεί την ιστορία του, διότι μία τροποποίηση θα μπορούσε να οδηγήσει σε ένα κρυμμένο σφάλμα, το οποίο θα ανακαλυπτόταν μόνο αργότερα, για παράδειγμα, ενώ η προτέρα έκδοση χρειάζεται να ανακτηθεί, τουλάχιστον για να αναλυθεί η ρίζα του προβλήματος. Αν το project αναπτύσσεται από πολλά άτομα, ο συγγραφέας της κάθε αλλαγής θα πρέπει και να καταγραφεί, για τους ίδιους λόγους που εξηγήσαμε παραπάνω. Εάν κυκλοφορούν διάφορες εκδόσεις ενός project, τότε πρέπει να ξέρουμε ακριβώς ποιές εκδόσεις της κάθε ενότητας (*module*) αποτελούν μέρος της συνολικής εκδόσεως. Συχνά, ένα project κρατά μία σταθερή και μία πειραματική έκδοση: αμφότερες πρέπει να συντηρούνται, να αποσφαλαματώνονται και να διορθώνονται τα λάθη που μεταφέρονται από τη μία έκδοση στην άλλη. Αυτό μπορεί να γίνεται με την αποθήκευση και το σωστό ετικετάρισμα της κάθε μεμονωμένης έκδοσης και υποέκδοσης, πράγμα που εθεωρείτο γενικά ένα υπερβολικό κόστος, αν και με τους σύγχρονους δίσκους αυτό ισχύει όλο και λιγότερο. Αυτό που κάνει, κανονικά, ένα Σύστημα ελέγχου πηγαιού κώδικα (*Source control system*), γνωστό και σαν Σύστημα διαχείρισης εκδόσεων (*Version management system*), είναι να αποθηκεύει όλο το ιστορικό των αρχείων, υπό μορφή ενός συνόλου αλλαγών, προς σύγκριση με μία έκδοση αναφοράς, συνήθως την πλέον πρόσφατη, και για λόγους αποτελεσματικότητας, προσαρτά επίσης και μία ετικέτα (*label*) σε κάθε μεμονωμένη αλλαγή, με όλα τα απαραίτητα μεταδεδομένα (*metadata*).

Χρειαζόμαστε, όμως, και ένα σύστημα με τέτοια χαρακτηριστικά που να χρησιμεύει για την αρμονική συνεργασία πολλών προγραμματιστών, χωρίς να χώνεται ο ένας μέσα στα πόδια του άλλου και χωρίς να εμποδίζεται η αμοιβαία πρόοδος. Συνεπώς, πρέπει να είμαστε σε θέση να επιτρέπουμε σε πολλούς προγραμματιστές να εργάζονται ταυτόχρονα, αλλά διατηρώντας και έναν έλεγχο. Αυτός ο έλεγχος μπορεί να είναι αισιόδοξος ή απαισιόδοξος.

Με τον απαισιόδοξο έλεγχο, ένας προγραμματιστής μπορεί να κρατά μερικά αρχεία για τον εαυτό του, μέχρι να τα βελτιώσει, για ένα διάστημα κατά το οποίο κανείς άλλος δεν μπορεί να τα αγγίξει. Αυτό μπορεί να είναι πολύ ασφαλές, αλλά θα εμποδίσει άλλους προγραμματιστές και, εν τέλει, μπορεί να καθυστερήσει το project, ειδικά αν ο προγραμματιστής που κλείδωσε τα αρχεία είναι απασχολημένος με άλλα πράγματα, είτε αν απλά ξέχασε εντελώς να ασχοληθεί με αυτά. Επιτρέποντας και σε άλλους να προχωρούν, μπορεί να είναι πιο δυναμικό, αλλά είναι και πιο επικίνδυνο, αφού μπορεί να οδηγήσει σε ασύμβατες μεταξύ τους τροποποιήσεις.

Ένα αισιόδοξο σύστημα ελέγχου επιτρέπει να σημειωθεί πρόοδος, αλλά μας προειδοποιεί όταν θα σημειώνονται συγκρούσεις (*conflicts*) και μας παρέχει τα εργαλεία για να τις επιλύουμε.

8.5.1. CVS (*Concurrent Version System/ Σύστημα Διαχείρισης Ταυτόχρονων Εκδόσεων*)

Το CVS είναι ένα αισιόδοξο σύστημα διαχείρισης πηγαιού κώδικα, που σχεδιάστηκε κατά τα τέλη

δεκαετίας του 1980 και χρησιμοποιείται από τη συντριπτική πλειοψηφία των ελεύθερων projects (*Concurrent Version System* [20], *Open source code development with CVS*, 2nd edition [113], *Version Management with CVS* [95]). Χρησιμοποιεί ένα κεντρικό αποθετήριο, προσβάσιμο μέσω ενός συστήματος πελάτη/διακομιστή (*client/server*). Ο διαχειριστής του συστήματος αποφασίζει ποιος θα έχει πρόσβαση στο αποθετήριο, ή σε ποια ακριβώς μέρη του αποθετηρίου, αν και, κανονικά, άπαξ και ένας developer γίνει δεκτός στον κύκλο της απόλυτης εμπιστοσύνης, τότε θα έχει πρόσβαση στα πάντα. Η ανώνυμη πρόσβαση περιλαμβάνει μόνο την “*read-only*” δυνατότητα, και μπορεί να παραχωρηθεί σε όλους.

Ο ανώνυμος συνεργάτης

Το ανώνυμο CVS είναι ένα εργαλείο ζωτικής φύσεως για την υλοποίηση της έννοιας της “πρώιμης και συχνής έκδοσης” (“*release early and often*”) που είχε προτείνει ο Eric Raymond. Όποιος από τους χρήστες αγχώνεται για να δοκιμάσει την πιο πρόσφατη έκδοση ενός προγράμματος, μπορεί να την αντλήσει (*extract*) από το αποθετήριο του CVS, να ανακαλύψει bugs και να τα αναφέρει, ακόμη και να υποβάλλει και διορθωτικά patches. Μπορεί, επίσης, να εξετάσει ολόκληρο το ιστορικό της ανάπτυξης.

Ας δούμε λίγο τους μηχανισμούς. Ας πάρουμε το παράδειγμα ενός προχωρημένου χρήστη που θέλει να αντλήσει την πιο πρόσφατη έκδοση του module “mod”, από ένα αποθετήριο που είναι προσβάσιμο με ανώνυμο τρόπο, από το progs.org, μέσα στο φάκελλο /var/lib/cvs και από τον διακομιστή πρωτοκόλλου pserver. Την πρώτη φορά θα δηλώσει την πρόθεσή του να μπει:

```
cvs -d:pserver:anonymous@progs.org:/var/lib/cvs login
```

Αν του ζητηθεί ένας κωδικός εισόδου, τότε αυτός θα είναι “*anonymous user*” (συνήθως το carriage return/ αυτόματη επιστροφή στη νέα γραμμή), και θα καταγραφεί σε ένα τοπικό αρχείο (παρά το ότι αυτή η διαδικασία δεν είναι πραγματικά απαραίτητη για την ανώνυμη πρόσβαση, το πρόγραμμα θα διαμαρτυρηθεί αν δεν υπάρχει το αρχείο με τον κωδικό εισόδου). Κατόπιν, είναι σημαντικό να ληφθεί το πρώτο αντίγραφο του αρθρώματος (*module*):

```
cvs -d:pserver:anonymous@progs.org:/var/lib/cvs co mod
```

Αυτό θα δημιουργήσει ένα φάκελλο “mod” με όλα τα αρχεία και τους φακέλλους της ενότητας (*module*), καθώς και με μερικά μεταδεδομένα (περιέχονται σε υποφακέλλους με το όνομα “CVS”), πράγμα που θα επιτρέψει, μεταξύ άλλων, να μην είναι υποχρεωτικό να επαναλαμβάνονται οι πληροφορίες που έχουν ήδη δοθεί.

Ο προχωρημένος μας χρήστης θα εισαγάγει το νεοδημιούργητο φάκελλο, θα δημιουργήσει το πακέτο και θα το τεστάρει:

```
cd mod

./configure

make

make install...
```

Όταν θελήσει να πάρει τη νεότερη έκδοση, θα πρέπει απλώς να κάνει ενημέρωση (update) του αντιγράφου του μέσα στο mod.

```
cd mod  
  
cvs update  
  
./configure  
  
make  
  
make install...
```

Αν βρει κάποιο bug, μπορεί να το διορθώσει επί τόπου και μετά να στείλει ένα patch μέσω e-mail, προς το συντηρητή του προγράμματος (είτε σε προσωπικό επίπεδο, είτε προς τη mailing list):

```
cvs diff -ubB | mail -s "My patches" mod-maint@progs.org
```

Ο συνήθης developer

Ο συνήθης developer θα έχει ένα λογαριασμό στο διακομιστή. Μπορεί να χρησιμοποιεί τον ίδιο μηχανισμό και το ίδιο πρωτόκολλο με τον ανώνυμο χρήστη, αντικαθιστώντας το “anonymous” με το όνομα του λογαριασμού του.

Σημείωση

Για λόγους ασφαλείας, για τους λογαριασμούς που έχουν δικαίωμα εγγραφής (*write permissions*), συνήθως χρησιμοποιείται το ssh, επειδή παρέχει ένα πιστοποιημένο (*authenticated*) και κρυπτογραφημένο (*encrypted*) κανάλι.

Μόλις έχει ένα λειτουργικό αντίγραφο του αρθρώματος (*module*), μπορεί να κάνει τις απαραίτητες αλλαγές, και όταν νομίζει ότι αυτές ολοκληρώθηκαν, τότε θα υποβάλλει (θα “κάνει *commit*”) τις αλλαγές στο αποθετήριο. Για παράδειγμα, αν τροποποιήσει τα αρχεία `part.h` και `part.c`, θα τα υποβάλλει ως εξής:

```
cvs ci part.h part.c
```

Πριν περατώσει αυτή την ενέργεια, το CVS θα ζητήσει μία εξήγηση σχετικά με το τι ακριβώς έκανε, η οποία θα επισυναφθεί και στα δύο αρχεία καταγραφής. Ο αύξων αριθμός αναθεώρησης του κάθε αρχείου θα μεγαλώσει, επίσης, κατά μία μονάδα. Αυτός ο αριθμός ταυτοποιεί την κάθε σημαντική στιγμή στην ιστορία ενός αρχείου και μπορεί να χρησιμοποιηθεί για να ανακτηθεί κάθε ένα από αυτά τα στιγμιότυπα.

Πότε θα πρέπει ένας developer να “κάνει commit”; Αυτό είναι ένα θέμα μεθοδολογίας, πάνω στο οποίο τα μέλη του project θα πρέπει να συμφωνήσουν, αλλά είναι προφανές ότι οι αλλαγές που δεν έχουν μεταγλωττισθεί δεν πρέπει να υποβάλλονται. Είναι, όμως, προτιμότερο να υπάρχει έστω και ένα ελάχιστο δοκιμαστικό τεστ. Σε πολλά projects απαιτείται και η έγκριση ενός επόπτη του project (ή του sub-project), ο οποίος θα ελέγξει εάν όντως η τροποποίηση απαιτείται.

Καθώς αναπτύσσεται η τροποποίηση, κάποιος μπορεί να έχει ήδη αλλάξει κάποια άλλα, ή ακόμη

και τα ίδια αρχεία. Συνεπώς, είναι προτιμότερο για τους developers να κάνουν συχνά update του δικού τους αντιγράφου (*ενημέρωση του cvs*). Αν τυχόν τροποποιήθηκαν άλλα αρχεία, μπορεί να έχει αλλάξει όλο το περιβάλλον και, με αυτή τη λογική, κάποιες δοκιμασίες που ολοκληρωνόταν με επιτυχία στο παρελθόν, μπορεί τώρα να αποτυγχάνουν. Αν τα ίδια αρχεία έχουν αλλάξει, μπορεί οι αλλαγές να έγιναν είτε σε τμήματα είτε σε ρουτίνες που εμείς δεν είχαμε πειράξει, ή μπορεί να έχουν γίνει πάνω στον κώδικα που εμείς τροποποιήσαμε. Στην πρώτη περίπτωση, δεν υπάρχει σύγκρουση (*conflict*), ή τουλάχιστον όχι εμφανώς και η διαδικασία τροποποίησης "συγχωνεύει" τη δική μας έκδοση, με εκείνη του αποθετηρίου, δημιουργώντας συνδυασμένα αρχεία, που θα περιλαμβάνουν όλες τις αλλαγές. Ειδικά, θα υπάρξει μία σύγκρουση (*conflict*), οπότε θα πρέπει να συζητήσουμε με τον developer ο οποίος έκανε τις άλλες τροποποιήσεις, ούτως ώστε να συμφωνήσουμε για την τελική έκδοση.

Για την καλύτερη ταυτοποίηση του κάθε τμήματος ενός project, είναι καλύτερα για το ίδιο το τμήμα αυτό να ενσωματώνει επάνω του κατ' ευθείαν τις σχετικές πληροφορίες αναθεώρησής του. Το CVS μπορεί να μαρκάρει τους πηγαίους κώδικες και τα διάφορα αντικείμενα, με αυτόματο τρόπο, υπό την προϋπόθεση ότι θα ακολουθούνται συγκεκριμένοι κανόνες. Για παράδειγμα, αν μέσα στο σχόλιο (*comment*) του Πηγαίου Κώδικα γράψουμε τη λέξη-κλειδί \$Id\$, τότε κάθε φορά που το αρχείο υποβάλλεται ("γίνεται commit") στο αποθετήριο, η λέξη αυτή θα αντικαθίσταται από μία σειρά ταυτοποίησης (*identification chain*) που θα δείχνει το όνομα του αρχείου, τον αριθμό αναθεώρησης⁸, την ημερομηνία και την ώρα που έγινε το commit και το δημιουργό του:

```
$Id: part.c,v 1.7 2003/07/11 08:20:47 joaquin Exp $
```

Αν συμπεριλάβουμε αυτή την λέξη-κλειδί σε μία συμβολοσειρά (*string*) του προγράμματος, όταν η συμβολοσειρά αυτή μεταγλωττισθεί, τότε θα φαίνεται το αντικείμενο και στο εκτελέσιμο αρχείο, κάνοντας εφικτό τον εντοπισμό του με ένα εργαλείο (*ident*).

⁽⁸⁾ Στο CVS οι αριθμοί αναθεώρησης έχουν κανονικά δύο συστατικά (το μείζον και το έλασσον), αλλά μπορούν να διαθέτουν και παραπάνω, πχ. 4, 6, κλπ.

Ο διαχειριστής

Οι διαχειριστές (*administrators*) είναι, προφανώς, υπεύθυνοι για το πιο περίπλοκο μέρος της συντήρησης αποθετηρίου. Πρέπει πχ.: να καταγράψουν το πρόγραμμα, να αποδώσουν δικαιώματα στους developers και να τους συντονίσουν, να βάλουν ετικέτες στις διάφορες εκδόσεις, κλπ.

Είναι κοινή πρακτική για όλα τα projects να έχουν μία σταθερή και μία πειραματική έκδοση. Για να το πετύχουν, δημιουργούν δύο κλάδους. Εκείνους που είναι αφιερωμένοι στην συντήρηση και διορθώνουν τα σφάλματα στο σταθερό κλάδο, ενώ οι νέες εξελίξεις, αντιθέτως, φορτώνονται στον πειραματικό κλάδο. Όταν ο πειραματικός κλάδος σταθεροποιηθεί, τότε το περιεχόμενό του μεταφέρεται στο σταθερό κλάδο, επιφέροντας πρώτα τις απαραίτητες διορθώσεις στο σταθερό κλάδο. Αυτή η διαδικασία είναι γνωστή ως συγχώνευση (*merging*), είναι μία ευαίσθητη διαδικασία και υποστηρίζεται από το CVS, αν και με έναν πρωτόγονο τρόπο. Αυτή η ιδέα μπορεί να επεκταθεί στην έννοια των πειραματικών κλάδων, που εξελίσσονται προς διάφορες κατευθύνσεις, οι οποίες ενδέχεται να καταλήξουν- ή όχι- με καλό τρόπο, αλλά που σε κάθε περίπτωση θα ενσωματωθούν πλήρως ή μερικώς στο σταθερό προϊόν, με τις κατάλληλες ενέργειες συγχώνευσης, εκτός κι' αν πρόκειται για προσπάθειες χωρίς ικανοποιητικό αποτέλεσμα.

Ένα δικαίωμα που μας δίνει το Ελεύθερο Λογισμικό είναι να μπορούμε να τροποποιούμε ένα πρόγραμμα για ιδιωτική μας χρήση. Αν και είναι επιθυμητό να συνεισφέρουμε τις οποιεσδήποτε

βελτιώσεις προς την κοινή δεξαμενή, συχνά οι αλλαγές που θέλουμε να κάνουμε μπορεί να είναι πολύ εξειδικευμένες ή χωρίς ενδιαφέρον για το ευρύ κοινό. Εμείς, παρ' όλα ταύτα, επιθυμούμε την ενσωμάτωση της εξέλιξης αυτής μέσα στο αρχικό πρόγραμμα. Αυτό μπορεί να γίνει με ένα ειδικό είδος διακλάδωσης και συγχώνευσης (κλάδοι πωλητών/ *vendor branches*).

Ο διαχειριστής μπορεί επίσης να διευκολύνει το συντονισμό της ομάδας μέσω αυτόματων μηχανισμών, όπως, για παράδειγμα, με τη δημιουργία μηνυμάτων e-mail όταν συμβαίνουν συγκεκριμένα πράγματα, όπως commits, ή επιβάλλοντας την εκτέλεση ορισμένων αυτόματων πράξεων πριν γίνει ένα commit, όπως κάποιοι αυτόματοι έλεγχοι του στυλ, μεταγλωττίσεις, ή tests.

8.5.2. Άλλα συστήματα διαχείρισης πηγών κώδικα

Παρ' όλο που το CVS αποτελεί ένα σύστημα ελέγχου εκδόσεων που έχει χρησιμοποιηθεί όσο πιο εκτεταμένα γίνεται, παρουσιάζει κάποια αξιοσημείωτα μειονεκτήματα:

- 1) Το CVS δεν υποστηρίζει αλλαγές ονόματος αρχείων, ή αλλαγές φακέλλου, ή των μεταδεδομένων (ιδιοκτήτης, δικαιώματα, κλπ.) ή συμβολικούς συνδέσμους (*symbolic links*).
- 2) Επειδή αποτελεί την εξέλιξη ενός συστήματος ελέγχου εκδόσεων για μεμονωμένα αρχεία, είναι φυσικό να μην υποστηρίζει τον έλεγχο εκδόσεων για ολόκληρες ομάδες.
- 3) Το CVS δεν υποστηρίζει ομάδες συναφών αλλαγών (*sets of coherent changes*). Πραγματικά, η προσθήκη ενός μεμονωμένου χαρακτηριστικού, είτε η διόρθωση ενός σφάλματος, μπορεί να συνεπιφέρει τροποποίηση πολλών αρχείων. Αυτές οι αλλαγές θα πρέπει να γίνουν μία προς μία.
- 4) Η χρήση των κλάδων και των συγχωνεύσεων στο CVS είναι αρκετά περίπλοκη. Πράγματι, αν δημιουργήσουμε έναν πειραματικό κλάδο ενός project και θέλουμε να συμπεριλάβουμε τις αλλαγές που έγιναν στη σταθερή έκδοση, πρέπει να ξέρουμε επακριβώς ποιες αλλαγές έχουν ήδη γίνει και ποιες όχι, ούτως ώστε να μην ξανακάνουμε τα ίδια πράγματα πολλές φορές.
- 5) Το CVS εξαρτάται από ένα κεντρικό διακομιστή, και αν και μπορούμε να δουλέψουμε και χωρίς σύνδεση, εν τούτοις χρειαζόμαστε μία σύνδεση για να δημιουργούμε εκδόσεις, να τις συγκρίνουμε και να τις συγχωνεύουμε.
- 6) Χωρίς τη βοήθεια χωριστών εργαλείων, το CVS δεν δημιουργεί το αρχείο καταγραφής αλλαγών ενός αρχείου (*file changelog*), που θα δείχνει την συνολική ιστορία των αλλαγών ενός project.
- 7) Το CVS δεν υποστηρίζει πολύ καλά τα projects που έχουν πολύ μεγάλο αριθμό αρχείων, όπως για παράδειγμα το project του πυρήνα Linux.

Και όμως, υπάρχουν άλλα ελεύθερα συστήματα που επιλύουν πολλά από τα προβλήματα αυτά. Θα μπορούσαμε να υπογραμμίσουμε το διάδοχο του CVS, για τον οποίο ήδη μιλήσαμε, το Subversion (<http://subversion.tigris.org>) [62], (<http://svnbook.red-bean.com/>) [96], που σαφώς επιλύει τα βασικά προβλήματα του CVS και μπορεί να κάνει χρήση των επεκτάσεων HTTP (WebDAV) για να προσπερνά τις επιθετικές πολιτικές ασφαλείας.

Σημείωση

Κατά το 2007, το Subversion είναι, ήδη, ο καθαρός διάδοχος του CVS και πολλά projects Ελεύθερου Λογισμικού μετέβησαν σε αυτό.

Το μοντέλο ανάπτυξης που βασίζεται σε ένα κεντρικό αποθετήριο, αν και κατάλληλο για συνεργατική χρήση, δεν ικανοποιεί όλες τις προσδοκίες, αφού η ικανότητα δημιουργίας των δικών μας κλάδων ανάπτυξης εξαρτάται, από τη μία μεριά, από την προσβασιμότητα αλλά και την καλή λειτουργία του διακομιστή και, από την άλλη μεριά, από τους διαχειριστές του διακομιστή αυτού. Ενίοτε, απαιτούνται διάσπαρτα αποθετήρια που επιτρέπουν στον καθένα να έχει ένα αποθετήριο, με ένα ιδιωτικό ή δημόσιο κλάδο που να μπορεί να συγχωνευθεί ή όχι με τον επίσημο κλάδο.

Αυτός είναι ο τρόπος με τον οποίο δουλεύει το GNU arch (*Arch Revision Control System*) [8] ή το bazaar (*Bazaar GPL Distributed Version Control Software*) [12], όπως επίσης και το ιδιοταγές σύστημα BitKeeper (*Bitkeeper Source Management*) [14], που επέλεξε ο Linus Torvalds για τη συντήρηση του Linux, από τον Φεβρουάριο του 2002, αφού κατά την άποψή του δεν υπήρχε ένα κατάλληλο ελεύθερο εργαλείο. Ελέγχθη ότι με τη χρήση του Bitkeeper διπλασιάστηκε ο ρυθμός εξέλιξης του Linux. Παρ' όλα αυτά, η απόφαση αυτή δέχθηκε βαριά κριτική λόγω της ιδιοταγούς φύσεως του προγράμματος, υπό μία άδεια που επέτρεπε στα ελεύθερα projects να το αποκτήσουν δωρεάν, υπό την προϋπόθεση ότι όλα τους θα υποβάλλουν τις αλλαγές με τα μεταδεδομένα τους σε έναν δημόσιο διακομιστή που θα τους υποδεικνύουν οι ιδιοκτήτες και θα είναι προσβάσιμος στον καθέναν, ανά πάσα στιγμή, υπό τον όρο ότι το αδειοδοτούμενο project δεν θα προσπαθήσει να αναπτύξει ένα διαφορετικό και ανταγωνιστικό σύστημα ελέγχου πηγαίου κώδικα. Ήταν ακριβώς η απόπειρα ανάπτυξης ενός συμβατού αλλά ελεύθερου προϊόντος εκ μέρους ενός υπαλλήλου της ίδιας εταιρείας όπου εργαζόταν και ο Linus Torvalds, που πυροδότησε την αλλαγή στο σύστημα διαχείρισης πηγαίου κώδικα. Ο Linus ανέπτυξε γρήγορα ένα προσωρινό αντικαταστάτη, το git (*"Git manual page"*) [218], που έγινε, επίσης τάχιστα, το οριστικό σύστημα, καθώς συγκέντρωνε όλη την εμπειρία της συνεργασίας από τον κόσμο του Linux και της αποκεντρωμένης ανάπτυξης: υποστηρίζει τα projects μεγάλου μεγέθους με έναν αποκεντρωμένο τρόπο, διευκολύνοντας σε μεγάλο βαθμό την ανάπτυξη των νεόκοπων κλάδων και τη συγχώνευσή τους με άλλους, είτε και με τον κυρίως κλάδο, μέσω κρυπτογραφημένων μηχανισμών ασφαλείας που αποτρέπουν την αλλοίωση των αρχείων καταγραφής (log). Από τον Απρίλιο του 2005, το Linux συντηρείται με την χρήση του git, ή των παραγώγων του (πχ. το cogito *"Cogito manual page"* [90]).

8.6. Γραπτή Τεκμηρίωση

Στον κόσμο του Ελεύθερου Λογισμικού, οι επεξεργαστές κειμένου WYSIWYG ("what you see is what you get"/ "ότι βλέπεις, ακριβώς εκείνο παίρνεις") και άλλα εργαλεία για σουίτες γραφείου, που σημείωσαν τόση επιτυχία σε άλλα περιβάλλοντα, εδώ χρησιμοποιούνται μόνο σπανίως, αν και υπάρχουν ήδη έτοιμα ελεύθερα εργαλεία, όπως το OpenOffice.org. Αυτό οφείλεται σε διάφορους παράγοντες:

- Είναι προτιμότερη η διατήρηση της γραπτής τεκμηρίωσης υπό στενό έλεγχο, και με συστήματα ελέγχου πηγαίου κώδικα, όπως το CVS, που αν και δέχονται δυαδικές μορφές αρχείου, προτιμούν τις διαφανείς, ανοιχτές μορφές αρχείων κειμένου, που είναι επεξεργάσιμα με έναν απλό επεξεργαστή κειμένου και η επεξεργασία τους γίνεται με εργαλεία που αναπτύχθηκαν για προγράμματα που μας επιτρέπουν να βλέπουμε εύκολα τις διαφορές ανάμεσα στις διάφορες εκδόσεις, να δημιουργούμε και να εφαρμόζουμε patches βασισμένα σε αυτές τις διαφορές και να εκτελούμε συγχωνεύσεις.

- Μερικές ελεύθερες Άδειες για τα έγγραφα, ειδικά η GFDL (βλέπε Κεφάλαιο 10.2.1), απαιτούν διαφανείς μορφές αρχείου, ιδιαίτερα διότι έτσι κάνουν πιο εύκολη τη δουλειά για εκείνους που θα συντάξουν παράγωγα έγγραφα.

- Τα εργαλεία WYSIWYG, σε γενικές γραμμές, δεν περιέχουν καμία άλλη πληροφορία πέρα από την σκέτη εμφάνιση, καθιστώντας πολύ δύσκολη, αν όχι εντελώς αδύνατη, την ταυτοποίηση των συγγραφέων, ή των τίτλων, ή ακόμη και την μετατροπή σε άλλου είδους μορφές αρχείου. Έστω και αν επιτρέπουν την μετατροπή σε άλλες μορφές αρχείου, αυτό συνήθως γίνεται διαδραστικά και είναι αδύνατον να αυτοματοποιηθεί (για παράδειγμα, με τη χρήση του “make”).

Σημείωση

Στο Unix τα πιο συνηθισμένα εργαλεία για αυτές τις δουλειές είναι το `diff`, το `diff3`, το `patch` και το `merge`.

- Σε γενικές γραμμές, οι εφαρμογές γραφείου παράγουν ευμεγέθη αρχεία, πράγμα που είναι ένα ανεπιθύμητο χαρακτηριστικό τόσο για τους developers όσο και για τα αποθετήρια.

Για όλους τους παραπάνω λόγους, οι ελεύθεροι προγραμματιστές, συνηθισμένοι στον προγραμματισμό και την μεταγλώττιση, προτιμούν τις διαφανείς μορφές αρχείου, σε πολλές περιπτώσεις τα απλά και αμιγή αρχεία κειμένου και σε πολλές άλλες, τις μορφές επεξεργάσιμου κειμένου.

Οι επεξεργάσιμες μορφές αρχείου δεν είναι, πάντως, και πάρα πολλές. Παραδοσιακά, στον κόσμο του Unix, η τεκμηρίωση των προγραμμάτων γινόταν στις μορφές αρχείου που ήταν αναμενόμενες για την οικογένεια επεξεργαστών τύπου *roff*, με μία ελεύθερη έκδοση (*GNU troff*) [37] από τον Norman Walsh. Εν τούτοις, αυτή η πρακτική εγκαταλείφθηκε σταδιακά, εκτός από την περίπτωση των παραδοσιακών σελίδων *manual*, αφού είναι σχεδόν υποχρεωτική η προετοιμασία των σελίδων *manual*, για τα πλέον βασικά εργαλεία του συστήματος. Πολλές από αυτές τις λεγόμενες σελίδες *manual*, διογκώθηκαν τόσο πολύ είναι σχεδόν λάθος να τις ονομάζουμε ακόμη *σελίδες*. Οπότε, κατέστη απαραίτητο να φτιαχτεί μία εναλλακτική μορφή αρχείου με υπερκείμενο, που να επιτρέπει την εμφάνιση δομημένων εγγράφων, με πίνακα περιεχομένων και πολλαπλές παραπομπές (cross-references) που να μπορεί κάποιος να ακολουθήσει. Το έργο GNU σχεδίασε τη μορφή αρχείου *texinfo* (*Texinfo - The GNU Documentation System*) [63] και την υιοθέτησε σαν το δικό του πρότυπο. Αυτή η μορφή αρχείου, επιτρέπει να δημιουργούμε πλοηγήσιμα έγγραφα με το εργαλείο *info tool* ή μέσα από τον editor *emacs*, και κατόπιν, να έχουμε ποιοτικές εκτυπώσεις εγγράφων με τη χρήση του επεξεργαστή κειμένου TeX, του Donald Knuth (*The TeXbook*) [156].

Αν απαιτείται, η μορφή αρχείου *texinfo* μπορεί να μεταφρασθεί σε πολυσέλιδη HTML και πολλοί προτιμούν να διαβάζουν τις πληροφορίες με έναν περιηγητή διαδικτύου (web navigator), αλλά με αυτό τον τρόπο χάνεται η δυνατότητα αναζήτησης λέξεων μέσα στο κείμενο. Αυτή είναι μία από τις παρενέργειες της δημοτικότητας της HTML, αφού τα προγράμματα περιήγησης διαδικτύου δεν εφαρμόζουν την έννοια του *πολυσέλιδου εγγράφου*, παρά το ότι υπάρχουν στοιχεία *διασύνδεσης* (link elements), που επιτρέπουν σε μεμονωμένα τμήματα του εγγράφου να διασυνδέονται.

Υπάρχει μία πιεστική ζήτηση για να μπορούμε να βλέπουμε σύνθετα έγγραφα το ίδιο εύκολα σαν να επρόκειτο για πλοηγήσιμες πολυσέλιδες ιστοσελίδες. Υπάρχουν και εκείνοι που γράφουν τεκμηρίωση σε LaTeX (*LaTeX Εγχειρίδιο χρηστών και Εγχειρίδιο αναφοράς*) [163], που είναι επίσης μία εφαρμογή TeX, πολύ διαδεδομένη στους επιστημονικούς κύκλους, πιο εκφραστική από το Texinfo και μετατρέψιμη σε πολυσέλιδες HTML με κάποια εργαλεία (*The LaTeX Web Companion*) [130], αρκεί να τηρούνται κάποιοι κανόνες. Πράγματι, οι εφαρμογές TeX είναι ομάδες μακροεντολών που συνδυάζουν τυπογραφικούς τελεστές (*operators*) πολύ χαμηλού επιπέδου, για να τους μετατρέπουν σε αφηρημένες γλώσσες, ικανές να δουλέψουν με έννοιες υψηλού επιπέδου (συγγραφέας, τίτλος, περίληψη, κεφάλαιο, παράγραφος, κλπ). Αν χρησιμοποιήσουμε μόνο τις βασικές μακροεντολές, τότε η μετατροπή είναι εύκολη. Επειδή, όμως, κανείς δεν μπορεί να αποτρέψει τη χρήση χαμηλού επιπέδου τελεστών (*operators*) και, επιπλέον, υπάρχουν τεράστιες

ποσότητες πακέτων με μακροεντολές, σαφώς πέραν των δυνατοτήτων των συντηρητών των εργαλείων μετατροπής, είναι πολύ δύσκολο να πετύχουμε καλές μετατροπές.

8.6.1. To DocBook

Το πρόβλημα προκύπτει από το γεγονός ότι δεν υπάρχει διάκριση ανάμεσα στο περιεχόμενο και την παρουσίαση, είτε στην TeX, ή στην *nroff*, αφού η αφαίρεση που περιγράφηκε είναι δομημένη σε επίπεδα. Αυτή η διάκριση γίνεται με τη χρήση εφαρμογών SGML (μία *standard*, γενικής χρήσης, γλώσσα σήμανσης/ *markup language*) [81] και XML (επεκτάσιμη γλώσσα σήμανσης) [224], όπου η παρουσίαση διευκρινίζεται με εντελώς διαφορετικά στυλ (*style sheets*). Σύντομα, άρχισαν να χρησιμοποιούνται πολύ απλές εφαρμογές SGML, όπως η *linuxdoc* και η *debiandoc*, αλλά λόγω της περιορισμένης εκφραστικής τους ικανότητας, επιλέχθηκε το DocBook. (*DocBook: the definitive guide*) [225].

Το DocBook είναι μία εφαρμογή SGML που αναπτύχθηκε αρχικά για την τεχνική τεκμηρίωση IT εγχειριδίων και τώρα διαθέτει και μία ποικιλία XML. Σήμερα, το DocBook είναι μία *standard* ελεύθερη μορφή αρχείου τεκμηρίωσης για πολλά projects (το Linux Documentation Project, το KDE, το GNOME, το Mandriva Linux, κλπ.) και αποτελεί ένα στόχο που προσπαθούν να φθάσουν οι υπόλοιποι (Linux, *BSD, Debian, etc).

Πάντως, το DocBook είναι μία περίπλοκη γλώσσα, που μαστίζεται από ετικέτες (*tags*), που σημαίνει ότι είναι χρήσιμο να υπάρχουν εργαλεία για να βοηθήσουν στο editing, έστω και αν είναι πολύ στοιχειώδη και δυσεύρετα. Ένα από τα πιο δημοφιλή εργαλεία αυτού του τύπου είναι το *psgml mode* του *emacs*. Είναι επίσης πολύ βαρύ στην επεξεργασία του και οι ελεύθεροι επεξεργαστές δεν έχουν καταφέρει ακόμη να παράγουν μία ιδιαίτερα ελκυστική ποιότητα εγγράφων.

8.6.2. Τα Wikis

Πολλοί βρίσκουν υπερβολικά περίπλοκο να γράψουν τεκμηρίωση με γλώσσες τόσο σύνθετες όπως το DocBook και με μηχανισμούς συνεργασίας σαν το CVS. Αυτός είναι ο λόγος που ένας νέος μηχανισμός συνεργασίας για online παρουσίαση εγγράφων μέσω του Διαδικτύου, έγινε τόσο δημοφιλής, με το όνομα wiki. Επινοήθηκε από τον Ward Cunningham ("*Wiki design principles*") [97]. Εφαρμόστηκε για πρώτη φορά το 1995 και, τώρα, χρησιμοποιείται ευρέως σε μία τεράστια ποικιλία εργασιών, για τη δημιουργία πολύ δυναμικών εγγράφων, που δεν σχεδιάστηκαν για εκτύπωση και συχνά είναι βραχύβιας διάρκειας (πχ., για την οργάνωση ενός Συνεδρίου).

Σε αντίθεση με το DocBook, ένα wiki προσφέρει μία πολύ απλή και συνοπτική γλώσσα σήμανσης, που θυμίζει αδρά την τελική παρουσίαση, χωρίς όμως να είναι ακριβώς η ίδια. Οι παράγραφοι, πχ., χωρίζονται από μία κενή γραμμή, τα στοιχεία που συνθέτουν έναν κατάλογο, όταν δεν είναι αριθμημένα, αρχίζουν με παύλα, ενώ, όταν είναι αριθμημένα, αρχίζουν με ένα 0, και τα κελιά των πινάκων χωρίζονται από κάθετες και οριζόντιες μπάρες.

Δεν υπάρχει καν η έννοια του "πλήρους εγγράφου": το wiki είναι περισσότερο σαν ένα σύνολο πολλών μικρών διασυνδεδεμένων εγγράφων, που δημιουργούνται, αν και όποτε απαιτείται, για να εξηγηθεί μία νέα έννοια, ή ένα νέο θέμα. Τα έγγραφα δημιουργούνται σχεδόν αυτόματα, καθώς το εργαλείο editing μας δείχνει σαφώς πως μπήκαμε σε μία νέα έννοια (με ένα *WikiName*, υπάρχουν σχεδόν πάντα δύο συνδεδεμένες λέξεις, με το πρώτο γράμμα κεφαλαίο). Σπανίως οποιοδήποτε wiki επιτρέπει υπερσυνδέσμους (*hyperlinks*) μέσα στην ίδια σελίδα.

Σε αντίθεση με το CVS, οποιοσδήποτε μπορεί να γράψει σε ένα *wiki*, αν και είναι προτιμότερο ο συγγραφέας να ταυτοποιήσει τον εαυτό του με μία εγγραφή. Όταν μπαίνουμε σε ένα *wiki*, μπορούμε να δούμε ότι όλες οι σελίδες έχουν ένα κουμπί που επιτρέπει να τις επεξεργαστούμε. Αν το πατήσουμε, ο περιηγητής θα μας ανοίξει μία φόρμα με τον πηγαίο κώδικα του εγγράφου, που είναι δυνατόν να αλλάξουμε. Δεν πρόκειται για μία τροποποίηση WYSIWYG, που θα αποθάρρυνε τον οποιονδήποτε που θα ήθελε να παρέμβει. Εδώ πρόκειται για κάτι αρκετά απλό, για οποιονδήποτε ενδιαφέρεται να τροποποιήσει τα έγγραφα με πολύ μικρή προσπάθεια.

Τα *Wikis* έχουν το δικό τους σύστημα ελέγχου εκδόσεων των εγγράφων, με τέτοιο τρόπο ώστε να είναι προσβάσιμες όλες οι αλλαγές, εμφανίζοντας ποιος τις έκανε και πότε. Μπορούν επίσης να συγκριθούν εύκολα. Επιπλέον, εμπεριέχουν συνήθως μηχανισμούς αναζήτησης, τουλάχιστον ως προς το όνομα της σελίδας και το περιεχόμενο σε λέξεις.

Κανονικά, ο αρχικός συγγραφέας μίας σελίδας θα θέλει να ξέρει τι αλλαγές έγιναν σε αυτή. Για να το καταφέρει, μπορεί να εγγραφεί στις αλλαγές ώστε να λαμβάνει ειδοποιήσεις σχετικά με αυτό, μέσω e-mail. Ενίοτε, το άτομο που βλέπει ένα έγγραφο δεν θα τολμήσει να αλλάξει τίποτα, αλλά μπορεί να προβεί σε ένα σχόλιο. Συνήθως, όλες οι σελίδες *wiki* έχουν ένα συνημμένο forum για τα σχόλια, που παρατίθεται στο τέλος του εγγράφου, και το οποίο μπορεί να χρησιμοποιήσει είτε ο αρχικός συγγραφέας, είτε οποιοσδήποτε άλλος που θα αναλάβει τον ρόλο του editor, για να αναθεωρήσει το αρχικό κείμενο, ενδεχομένως μετακινώντας φράσεις από τα σχόλια προς τις αντίστοιχα εδάφια του κειμένου.

Συμβουλή

Ο καλύτερος τρόπος για να κατανοήσει κανείς την έννοια του *wiki*, είναι να μπει σε κάποιο και να πειραματισθεί, σε μία σελίδα που σχεδιάστηκε γι' αυτόν ακριβώς το σκοπό και που, συνήθως, την αποκαλούν *SandBox*.

8.7. Διαχείριση των Σφαλμάτων (Bugs) και άλλα θέματα

Ένα από τα ισχυρά σημεία του μοντέλου της ελεύθερης ανάπτυξης είναι ακριβώς το γεγονός ότι η κοινότητα συνεισφέρει με τις αναφορές σφαλμάτων και θεωρεί σημαντικό να δίδεται σημασία σε αυτές τις αναφορές ή τις λύσεις. Αυτό προϋποθέτει έναν απλό μηχανισμό για την υποβολή αυτών των αναφορών, έτσι ώστε οι developers να λαμβάνουν επαρκή πληροφόρηση, με ένα συστηματικό τρόπο που να περιέχει όλες τις απαραίτητες λεπτομέρειες. Αυτές θα τις προσφέρει ο συνεργάτης, με μία εξήγηση για το τι ακριβώς συμβαίνει, το επίπεδο σοβαρότητας και την πιθανή λύση, είτε μέσω ενός αυτόματου μηχανισμού που θα καθορίζει, για παράδειγμα, την έκδοση του προγράμματος και το περιβάλλον μέσα στο οποίο λειτουργεί. Τα σφάλματα θα πρέπει επίσης να αποθηκεύονται σε μία βάση δεδομένων που να μπορεί κανείς να την συμβουλευθεί εύκολα, για να βλέπει αν ένα bug έχει ήδη αναφερθεί, διορθωθεί, τον βαθμό σοβαρότητας, κλπ.

Υπάρχουν διάφορα τέτοια συστήματα, με διαφορετικές φιλοσοφίες λειτουργίας. Μερικά λειτουργούν μέσω του Διαδικτύου, άλλα μέσω e-mail, με τη βοήθεια κάποιου ενδιάμεσου προγράμματος. Όλα διαθέτουν μία διεπαφή web για να μπορεί κανείς να τη συμβουλευθεί. Μερικά επιτρέπουν την υποβολή ανώνυμων αναφορών, ενώ άλλα απαιτούν ταυτοποίηση (μία έγκυρη διεύθυνση e-mail) για να προλαμβάνουν το θόρυβο. Αν και θα έλεγε κανείς ότι οι διαδικασίες μέσω του web φαίνονται να είναι οι πιο απλές που γίνεται, δεν καταφέρνουν όμως να πάρουν αυτόματη πληροφορία σχετικά με το περιβάλλον στο οποίο εμφανίστηκε το bug. Το σύστημα της Debian, πχ. προσφέρει προγράμματα σαν το `reportbug`, το οποίο αφού ζητήσει το όνομα του πακέτου που θέλουμε να αναφέρουμε, συμβουλευτεί το διακομιστή σφαλμάτων για τα bugs που έχουν ήδη αναφερθεί εκεί.

Αν καμία από τις υπάρχουσες αναφορές δεν σχετίζεται με το πρόβλημά μας, τότε θα μας ζητηθεί να δώσουμε μία περιγραφή του, του επιπέδου σοβαρότητας ("κρίσιμη", "βαριά", "σοβαρή", "σημαντική", "δεν μπορεί να αναπαραχθεί από τον πηγαίο κώδικα", "φυσιολογικό", "έλασσον" ή "πρόταση") και κάποια ετικέτα σχετικά με την κατηγορία του (πχ.: "ασφάλεια"). Κατόπιν, αν επιβεβαιώσουμε το αίτημα, θα εντοπίσει την έκδοση του πακέτου και τα πεκέτα από τα οποία εξαρτάται, πέραν της έκδοσης του πυρήνα και της αρχιτεκτονικής. Προφανώς, γνωρίζει τη διεύθυνση e-mail και, έτσι, στέλνει στο σωστό site, μία αναφορά όπως η παρακάτω:

```
Package: w3m-ssl
Version: 0.2.1-4
Severity: important
```

Μετά τη φόρτωση μίας σελίδας που περιέχει σύνθετους πίνακες, επί δεκάδες φορές, το w3m είχε χρησιμοποιήσει όλη τη φυσική μνήμη και άρχισε το **thrashing**. Αυτή είναι μία μηχανή Alpha.

```
--System Information

Debian Release: testing/unstable

Kernel Version: Linux romana 2.2.19 #1 Fri Jun 1 18:20:08 PDT 2001
alpha unknown

Versions of the packages w3m-ssl depends on:

ii libc6.1 2.2.3-7 GNU C Library: Shared libraries and Timezone
data
ii libgc5 5.0.alpha4-8 Conservative garbage collector for C
ii libgpm1 1.19.3-6 General Purpose Mouse Library [libc6]
ii libncurses5 5.2.20010318-3 Shared libraries for terminal
handling
ii libssl0.9.6 0.9.6a-3 SSL shared libraries
ii w3m 0.2.1-2 WWW browsable pager with tables/frames support
```

Αυτό το μήνυμα δημιουργεί έναν αριθμό σφάλματος (*bug*) που επιστρέφει σε μας, αποστέλλεται στον συντηρητή του πακέτου και αποθηκεύεται στη βάση δεδομένων. Όταν το bug επιλυθεί, θα λάβουμε πάλι μία ειδοποίηση. Ορίζεται μία διεύθυνση e-mail για το κάθε bug, που μπορεί πχ. να χρησιμοποιηθεί για την παροχή επιπρόσθετων πληροφοριών. Μπορούμε να ανατρέξουμε στη βάση δεδομένων με τα bug <http://bugs.debian.org> ανά πάσα στιγμή.

Μερικές φορές τα συστήματα επίβλεψης των bugs διαθέτουν μηχανισμούς για να αναθέτουν σε κάποιον να τα επιλύει και να ορίζουν και μία προθεσμία επίλυσης. Υπάρχουν και άλλα θέματα, όπως οι εργασίες σε εκκρεμότητα, οι βελτιώσεις που ζητήθηκαν, οι μεταφράσεις, κλπ, που απαιτούν παρόμοιους μηχανισμούς διαχείρισης.

Με το Ελεύθερο Λογισμικό δεν μπορούμε σε γενικές γραμμές να χρησιμοποιούμε άτεγκτους μηχανισμούς για τη διαχείριση των εργασιών που έχει να φέρει εις πέρας ο κάθε developer. Σε τελευταία ανάλυση, πολλοί συνεργάτες είναι εθελοντές και δεν μπορεί να υποχρεωθούν να κάνουν κάτι. Εν τούτοις, μπορούμε να προβούμε σε ανάθεση εργασιών και μπορούμε να περιμένουμε μέχρι κάποιος να γραφεί στο σύστημα και να αναλάβει να τις φέρει εις πέρας, εντός μίας δηλωμένης προθεσμίας. Ως προς το κατά πόσο υπάρχει δυνατότητα ελέγχου σχετικά με το τι ορισμένοι άνθρωποι μπορούν να κάνουν ή όχι, είναι πάντοτε προτιμότερο να ελέγχονται όλες οι εργασίες που πρέπει να γίνουν, ποιος και από τι εξαρτάται, το βαθμό σημασίας τους, και ποιος δουλεύει μαζί τους. Πολλά σημαντικά projects διαχειρίζονται αυτές τις πτυχές με τη χρήση του Bugzilla (*The Bugzilla guide*) [89] ή των παραγώγων του.

Μερικές φορές, κάποιος που δουλεύει σε ένα project, μπορεί να ανακαλύψει ένα bug σε ένα διαφορετικό project, από το οποίο εξαρτάται η δική του δουλειά, αλλά αυτό έχει ένα διαφορετικό σύστημα διαχείρισης των bugs σε σχέση με εκείνο στο οποίο έχει συνηθίσει ο ίδιος. Αυτό ισχύει ιδιαίτερα για χρήστες διανομών που θέλουν να χρησιμοποιήσουν ένα μεμονωμένο εργαλείο για να αναφέρουν και να παρακολουθήσουν την επίλυση ενός bug. Για να διευκολυνθεί η υποβολή αναφορών και η παρακολούθηση αυτών των bugs, μπορεί να είναι σκόπιμο να συνενώσει κανείς τη χρήση διαφορετικών συστημάτων, όπως κάνει το εργαλείο *Malone* (*The Malone Bug Tracker*) [47].

8.8. Υποστήριξη για άλλες αρχιτεκτονικές

Η ελάχιστη απαραίτητη υποστήριξη που απαιτείται όταν δουλεύουμε με ένα φορητό πρόγραμμα είναι η δυνατότητα πρόσβασης σε μαζικές υπηρεσίες μεταγλώττισης (*compilation farms*), που επιτρέπουν στο πρόγραμμα να μπορεί να μεταγλωττισθεί σε διαφορετικές αρχιτεκτονικές και λειτουργικά συστήματα. Το SourceForge, για παράδειγμα, (βλέπε κεφάλαιο 8.9.1) πέραν των Solaris και Mac OS/X, προσέφερε για ένα διάστημα περιβάλλον Debian GNU/Linux, για τις αρχιτεκτονικές Intel x86, DEC Alpha, PowerPC και SPARC.

Είναι, επίσης, χρήσιμο να μπορεί κανείς να δοκιμάσει και όχι απλώς να μεταγλωττίσει το πρόγραμμα σε εκείνα τα περιβάλλοντα. Αυτού του είδους η υπηρεσία απαιτεί περισσότερους πόρους και περισσότερο χρόνο εκ μέρους του διαχειριστή. Η υπηρεσία μεταγλώττισης μπορεί να είναι ήδη προβληματική, διότι χρειαζόμαστε συνήθως να προσφέρουμε περιβάλλοντα μεταγλώττισης σε περισσότερες γλώσσες, με έναν μεγάλο αριθμό βιβλιοθηκών. Εάν, λοιπόν, ο στόχος μας είναι να δοκιμάσουμε κάποιο πρόγραμμα, οι δυσκολίες αυξάνουν εκθετικά, όχι μόνο επειδή είναι πολύ δύσκολο να διαθέτουμε τους απαραίτητους πόρους, αλλά και για λόγους ασφαλείας, πράγμα που ενδεχομένως θα κάνει εξαιρετικά περίπλοκη τη διαχείριση τέτοιων συστημάτων. Εν τούτοις, υπάρχουν μερικές υπηρεσίες μαζικής μεταγλώττισης (*compilation farms*), με standard εγκαταστάσεις διαφόρων αρχιτεκτονικών, που επιτρέπουν να δοκιμάσουμε κάποια πράγματα.

Οι παραπάνω αναφερόμενες δημόσιες “farms” είναι, συνήθως, ένα είδος υπηρεσίας που προϋποθέτει χειροκίνητη (*manual*) χρήση. Ο προσκεκλημένος developer αντιγράφει τα αρχεία του σε ένα από αυτά τα μηχανήματα, τα μεταγλωττίζει και δοκιμάζει το αποτέλεσμα. Μάλλον θα χρειασθεί να επαναλαμβάνει την διαδικασία αυτή κάπου-κάπου, ειδικά πριν την κυκλοφορία μίας σημαντικής έκδοσης του προγράμματος. Θα ήταν πολύ πιο ενδιαφέρον τόσο για τις μεταγλωττίσεις, όσο και για την εκτέλεση των δοκιμών παλινδρόμησης (*regression tests*), να γίνονται συστηματικά, με αυτόματο τρόπο, για παράδειγμα κάθε βράδυ, όταν έχουν επέλθει αλλαγές στον πηγαίο κώδικα. Αυτός είναι ο τρόπος, με τον οποίο λειτουργούν κάποια σημαντικά projects, που παρέχουν τη δική τους υποδομή προς τους εξωτερικούς developers και το αποκαλούν “tinderbox”. Αυτή είναι και η περίπτωση του Mozilla, που χρηματοδοτείται από το Netscape, του οποίου το tinderbox

(<http://www.mozilla.org/tinderbox.html>) [50] έχει μία διεπαφή web με τα αποτελέσματα της μεταγλώττισης και των διαφόρων δοκιμών όλων των επί μέρους συστατικών του Navigator, σε όλες τις αρχιτεκτονικές στις οποίες πρόκειται να λειτουργήσει. Αυτή η διεπαφή σχετίζεται στενά με το CVS και δείχνει τα αποτελέσματα διαφόρων καταστάσεων (ανάμεσα στα διάφορα commits), εντοπίζοντας εκείνο που είναι υπεύθυνο για τα bugs και διευκολύνοντας την πρόοδο της αντιμετώπισής τους, προσπερνώντας το πρόβλημα μέχρι την επίλυσή του. Τα tinderboxes χρησιμοποιούνται, επίσης, από τα projects του OpenOffice και του FreeBSD, αλλά και από άλλα.

8.9. Ιστοσελίδες για υποστήριξη της ανάπτυξης

Οι ιστοσελίδες υποστήριξης της ανάπτυξης προσφέρουν, κατά τρόπο λιγότερο ή περισσότερο ολοκληρωμένο, όλες τις προαναφερθείσες υπηρεσίες συν μερικές επιπλέον, που δίνουν την δυνατότητα να μπορεί κανείς να ψάξει μέσα σε ένα project ανά κατηγορίες, καθώς και να τα κατατάξει με βάση μερικές απλές παραμέτρους δραστηριότητας. Αυτό γλυτώνει τον developer από το να έχει να στήσει και να διαχειρισθεί μία ολόκληρη υποδομή για τη συνεργασία, επιτρέποντάς του να επικεντρωθεί στο ίδιο το project.

8.9.1. To SourceForge

Αναφορικά με αυτού του είδους τις υπηρεσίες, μία από τις πρώτες που καθιερώθηκαν, και από τις πιο δημοφιλείς, είναι το SourceForge (<http://sourceforge.net>) [61]: το διαχειρίζεται το OSDN (*Open Software Development Network/ Δίκτυο Ανάπτυξης Ανοικτού Λογισμικού*), μία θυγατρική της VA Software, που τον Μάρτιο του 2007 φιλοξενούσε παραπάνω από 144.000 projects. Είναι δομημένο γύρω από μία ομάδα προγραμμάτων με το ίδιο όνομα και τα οποία, μέχρι και την έκδοση 2 ήταν Ελεύθερο Λογισμικό.

Το SourceForge, σαν πρωτότυπο αυτού του τύπου ιστοσελίδων, προσφέρει μία διεπαφή web ή πύλη (portal) παγκόσμιας πρόσβασης (<http://sourceforge.net/>) και ένα subportal ανά project (<http://project.sourceforge.net>). Η παγκόσμια διεπαφή δείχνει νέα, διαφημίσεις, links, και μία πρόσκληση για να εγγραφεί κανείς ως μέλος ή την δυνατότητα εισόδου, εφ' όσον είναι ήδη μέλος. Για να συνεργασθεί κανείς σε αυτή την ιστοσελίδα, είναι προτιμότερο να γίνει μέλος, ενώ αυτό είναι υποχρεωτικό αν θέλουμε να δημιουργήσουμε ένα νέο project είτε να συμμετάσχουμε σε ένα ήδη υπάρχον. Εάν, αντιθέτως, κάποιος θέλει απλώς να είναι θεατής, τότε δεν είναι απαραίτητο, και ως θεατής μπορούμε να δούμε ποια είναι τα projects με την πιο δραστήρια ανάπτυξη ή για τα οποία γίνονται τα περισσότερα downloads, είτε να ψάξουμε για projects ανά κατηγορία, ή με βάση ένα περιγραφικό όρο, και θα μας τα εμφανίσει κατά σειρά επιπέδου δραστηριότητας. Για κάθε project μπορούμε να δούμε μία περιγραφή, την κατάστασή του (*alpha*, *beta*, *production*), τις λεπτομέρειές του (γλώσσα προγραμματισμού, λειτουργικό σύστημα, περιοχή εφαρμογής, τύπος χρηστών, γλώσσα, τύπος αδείας...), bugs και διάφορα προβλήματα σε εκκρεμότητα, επίπεδα δραστηριότητας σε συνάρτηση με τον χρόνο..., ή τέλος και να το κατεβάσουμε. Μπορούμε επίσης να πάρουμε μέρος σε forums ή να υποβάλλουμε αναφορές για bugs, ακόμα και ανώνυμα, πράγμα που δεν συστήνεται ιδιαίτερα διότι μπορεί, για παράδειγμα, να μη λάβουμε ποτέ απάντηση.

Ο κάθε εγγεγραμμένος χρήστης μπορεί να ζητήσει την εγγραφή (*register*) ενός project, που οι διαχειριστές θα αποδεχθούν, αρκεί να συμμορφώνεται με τις πολιτικές της ιστοσελίδας, που όμως, στην περίπτωση του SourceForge είναι αρκετά φιλελεύθερες. Όταν θα έχει λάβει την απαραίτητη έγκριση, ο δημιουργός μπορεί να εγγράψει άλλους χρήστες ως επιπρόσθετους διαχειριστές ή ως developers, με δικαίωμα πρόσβασης και τροποποίησης των πηγών. Μετά την έγκριση, δεν ακολουθούν πολλοί περαιτέρω έλεγχοι στο project, πράγμα που σημαίνει ότι υπάρχουν πολλά ανενεργά projects.

Αυτό πάντως, δεν αποπροσανατολίζει τους χρήστες και τόσο πολύ, διότι οι αναζητήσεις των projects κατατάσσουν τα projects κατά σειρά επιπέδου δραστηριότητας και, άρα, τα projects με χαμηλό ή και ανύπαρκτο επίπεδο δραστηριότητας μόλις που εμφανίζονται. Αυτά τα projects διατρέχουν τον κίνδυνο διαγραφής εκ μέρους των ιδιοκτητών της ιστοσελίδας. Οι υπηρεσίες που το SourceForge προσφέρει σε ένα project, και που θα περιμέναμε να προσφέρονται και από οποιαδήποτε άλλη υπηρεσία, είναι οι εξής:

- Φιλοξενία για τις ιστοσελίδες της πύλης του project, στη διεύθυνση project.sourceforge.net, για να εμφανίζονται στο ευρύ κοινό. Αυτές οι σελίδες μπορεί να είναι στατικές ή δυναμικές (με CGI ή με PHP), στην οποία περίπτωση μπορούν να χρησιμοποιούν μία βάση δεδομένων (MySQL). Μπορεί κανείς να έχει άμεση πρόσβαση σε αυτές μέσω εντολών απομακρυσμένου αντιγράφου (*remote copy commands*) και μπορεί κανείς να το διαχειρισθεί με τη χρήση διαδραστικών συνεδριών απομακρυσμένου τερματικού (*remote terminal interactive sessions, SSH*).
- Προαιρετικά, έναν εικονικό διακομιστή (*virtual server*) που απαντά σε διευθύνσεις ενός άλλου domain, που ελήφθη ξεχωριστά, σαν το www.project.org ή το cvs.project.org.
- Όσα web forums ή/και mailing lists θεωρηθούν απαραίτητα, κατά τη γνώμη του διαχειριστή.
- Μία υπηρεσία διάδοσης νέων, όπου οι διαχειριστές αναγγέλλουν τις προόδους του project.
- Ιχνηλάτες (Trackers) για την υποβολή αναφορών για τα bugs και την παρακολούθησή τους, αιτήματα για υποστήριξη, για βελτιώσεις ή την ενσωμάτωση κάποιων patches. Οι διαχειριστές αποδίδουν στο θέμα ένα επίπεδο προτεραιότητας και αναθέτουν σε έναν developer να βρει τη λύση.
- Διαχειριστές εργασιών (*Task managers*), όμοιοι με τους trackers, που επιτρέπουν τον καθορισμό μίας σειράς εργασιών για τα sub-projects. Γι' αυτές τις εργασίες, εκτός από έναν βαθμό προτεραιότητας, ορίζεται και μία προθεσμία. Κάθε τόσο, οι developers στους οποίους ανετέθησαν, μπορούν να δείξουν τα ποσοστά ολοκλήρωσης που πέτυχαν.
- Ένα CVS ή Subversion, με δικαιώματα αρχικής πρόσβασης για όλους τους developers.
- Μία υπηρεσία για upload και download πακέτων λογισμικού. Καταγράφει τις εκδόσεις που έχουν μπει, όταν χρησιμοποιούνται, και τα ενδιαφερόμενα μέρη λαμβάνουν μία ειδοποίηση όταν αυτό συμβαίνει. Επιπλέον, το αρχικό upload περιλαμβάνει τη δημιουργία πολλαπλών αντιτύπων (replicas), σε παγκόσμιο επίπεδο, πράγμα που διευκολύνει την διανομή.
- Μία υπηρεσία για τη δημοσίευση εγγράφων σε μορφή HTML. Οποιοσδήποτε θα μπορεί να τα εγγράφει, αλλά θα είναι ορατά μόνο μετά την έγκριση από έναν διαχειριστή.
- Ένα αντίγραφο ασφαλείας για ανάκτηση σε περίπτωση ζημίας, όπως ένας χαλασμένος δίσκος, bugs που δεν προέρχονται από λάθος χρήστη, ή σε περίπτωση κατά λάθος διαγραφής ενός αρχείου.
- Έναν ενσωματωμένο μηχανισμό για δωρεές σε χρήστες, σε projects και στο Source-Forge.

Ένας πιστοποιημένος χρήστης θα έχει μία προσωπική σελίδα με όλες τις σχετικές πληροφορίες, όπως τα projects με τα οποία ο χρήστης συνδέθηκε, θέματα ή εργασίες που είναι σε εκκρεμότητα καθώς και αρχεία και forums που δήλωσε ότι θέλει να παρακολουθεί. Για να μη χρειάζεται να φροντίζει μόνος του την προσωπική του σελίδα, ο χρήστης θα λαμβάνει επιπλέον ειδοποιήσεις στην ηλεκτρονική του διεύθυνση σχετικά με όλα όσα επιθυμεί να ελέγχει.

8.9.2. Οι διάδοχοι του SourceForge

Το 2001, η VA Software κόντευε να χρεοκοπήσει, μέσα στη δίνη της κρίσης των εταιρειών dotcom. Κατόπιν, ανακοίνωσε μία νέα έκδοση του λογισμικού της για το SourceForge με μία μη-ελεύθερη έκδοση, σε μία απόπειρα να διασφαλίσει μία πηγή εσόδων, πουλώντας το προς τις εταιρείες, για τη δική τους εσωτερική ανάπτυξη. Συγχρόνως, εξάλειψε τους μηχανισμούς που επέτρεπαν σε ένα project να εγκαταλείπει και να μεταβαίνει σε κάποια άλλη ιστοσελίδα. Αμφότερες αυτές οι κινήσεις θεωρήθηκαν ως μία απειλή για τα χιλιάδες projects που φιλοξενούσε το SourceForge, υπό την έννοια ότι θα εγκλωβιζόταν στα χέρια μίας και μόνον εταιρείας, η οποία θα χρησιμοποιούσε την πλατφόρμα για την ανάδειξη μη-ελεύθερου λογισμικού. Αντιμέτωποι είτε με αυτή την πιθανότητα, ή με το ενδεχόμενο κλεισίματος της ιστοσελίδας, αναπτύχθηκαν παράγωγα της νέας έκδοσης και εμφανίστηκαν portals που βασιζόταν σε αυτή, ιδιαίτερα το Savannah (<http://savannah.gnu.org>) [57], αφιερωμένο στο GNU project και σε άλλα προγράμματα υπό Άδειες *copyleft*, ή το BerliOS (*BerliOS: The Open Source Mediator*) [13], που λειτούργησε ως ένα κοινό σημείο συνάντησης ανάμεσα στους developers Ελεύθερου Λογισμικού και τις εταιρείες. Αυτό, πάντως, ήταν απλώς ένα βήμα προς την κατεύθυνση της ανάπτυξης μίας κατανεμημένης και αναπαραγόμενης πλατφόρμας, όπου κανείς δεν θα διαθέτει την απόλυτη εξουσία πάνω στα διάφορα φιλοξενούμενα projects (*Savannah The Next Generation, 2001*) [98].

Ένα παράδειγμα συστήματος διαχείρισης των projects Ελεύθερου Λογισμικού είναι το Launchpad (<https://launchpad.net>) [43], που χρησιμοποιεί το Ubuntu για την ανάπτυξη της κάθε έκδοσης της διανομής αυτής. Το Launchpad δεν είναι ένα αποθετήριο πηγαίου κώδικα, αλλά σχεδιάστηκε με πιο πολύ για να προσφέρει υποστήριξη στην επίβλεψη του κώδικα, ατυχημάτων και μεταφράσεων. Για να το πετύχει, χρησιμοποιεί το εργαλείο Malone, για το οποίο κάναμε ήδη λόγο, και το οποίο επιτρέπει την ανακατεύθυνση των διάφορων συμβάντων, προς το κάθε ξεχωριστό αποθετήριο κώδικα των αντίστοιχων εμπλεκόμενων αρθρωμάτων (*modules*).

8.9.3. Άλλες ιστοσελίδες και προγράμματα

Φυσικά, αναπτύχθηκαν διάφορα συστήματα συνεργασίας και συνεχίζονται, και τώρα να αναπτύσσονται, ενώ μερικές εταιρείες έχουν ως βασικό επιχειρηματικό τους αντικείμενο την παροχή συντήρησης και τεχνικής υποστήριξης αυτού του είδους ιστοσελίδων. Για παράδειγμα, το Tigris project (<http://www.tigris.org/> : *Open Source Software Engineering Tools*) [64], το οποίο δεν συντηρεί απλώς διάφορα projects engineering Ελεύθερου Λογισμικού, αλλά χρησιμοποιεί και μία διαδικτυακή πύλη συνεργασίας (collaboration portal), το SourceCast (<http://sourcecast.org/>), που συντηρείται από μία εταιρεία παροχής υπηρεσιών, την CollabNet, που με τη σειρά της συντηρεί και ιστοσελίδες διαφόρων μεμονωμένων projects, όπως του OpenOffice.org. Οι διάφορες νεοαναδυόμενες ιστοσελίδες υιοθετούν και νέες μορφές ελεύθερου λογισμικού, σαν το GForge (<http://gforge.org>) [30], που χρησιμοποιείται από το Debian project (<http://alioth.debian.org>) [5]. Μπορούμε να δούμε μία αναλυτική σύγκριση πολλών ιστοσελίδων στο "*Comparison of free/open source hosting (FOSPhost) sites available for hosting projects externally from project owners*" [202].

9. Μελέτες Περίπτωσης (Case studies)

Το όνομα "GNU, που σημαίνει "Gnu's Not Unix", είναι το όνομα ενός συστήματος λογισμικού που είναι πλήρως συμβατό με περιβάλλον Unix, το οποίο συγγράφω για να μπορώ να το δίνω ελεύθερα σε όλους εκείνους που μπορούν να το χρησιμοποιούν. Πολλοί εθελοντές με βοηθούν σε αυτό το

εγγχείρημα. Υπάρχει μεγάλη ανάγκη για συνεισφορά σε χρόνο, χρήμα, προγράμματα και εξοπλισμό” γράφει ο Richard Stallman, στο “The GNU Manifesto” (1985)

Αυτό το κεφάλαιο προσφέρει μία πιο βαθιά μελέτη μερικών από τα πλέον ενδιαφέροντα projects Ελεύθερου Λογισμικού, από την άποψη της επίδρασης που είχαν στον κόσμο του Ελεύθερου Λογισμικού, των αποτελεσμάτων που πέτυχαν, των μοντέλων διαχείρισης, της ιστορικής εξέλιξης, κλπ. Σίγουρα, ο συνολικός αριθμός των projects που μπορούμε να συζητήσουμε εδώ, είναι κατά πολύ μικρότερος από τον συνολικό αριθμό των υπαρχόντων projects Ελεύθερου Λογισμικού (δεκάδες χιλιάδες), πράγμα που σημαίνει ότι αυτό το κεφάλαιο δεν θα πρέπει να εκληφθεί ως “πλήρες”, αλλά ούτε και θα μπορούσε ποτέ να ήτανε. Παρ' όλα αυτά, ελπίζουμε πως οι αναγνώστες, μετά την ανάγνωσή του, θα έχουν, τουλάχιστον, μία βασική κατανόηση για το πως οι γενικές θεωρίες που εξετάσαμε μέσα στο βιβλίο αυτό, έχουν τεθεί σε εφαρμογή στην πράξη.

Τα projects που επιλέξαμε ποικίλουν από εφαρμογές χαμηλότερου επιπέδου, εκείνες που αλληλεπιδρούν περισσότερο με το φυσικό σύστημα του υπολογιστή παρά με τον χρήστη, μέχρι και περιβάλλοντα εργασίας που σχεδιάστηκαν για τον τελικό χρήστη. Συμπεριλάβαμε και projects Ελεύθερου Λογισμικού που, επί της αρχής, δεν αποτελούν, αυστηρώς, έργα ανάπτυξης κώδικα. Αυτό ισχύει κυρίως για τις διανομές, που συνήθως χρησιμοποιούνται σαν ενοποιημένα συστήματα, καθώς παίρνουν κυρίως ένα εκτεταμένο, αλλά περιορισμένο, σύνολο ανεξάρτητων εφαρμογών και τις χρησιμοποιούν για να δημιουργήσουν ένα σύστημα, εντός του οποίου τα πάντα αλληλεπιδρούν αποτελεσματικά, περιλαμβανομένης και της επιλογής εγκατάστασης, ενημέρωσης και διαγραφής διαφόρων εφαρμογών, αναλόγως των επιθυμιών του χρήστη.

Τα projects κατώτερου επιπέδου που θα εξετάσουμε, είναι το Linux, ο πυρήνας των περισσότερων από τα σημερινά πιο δημοφιλή Ελεύθερα Λειτουργικά Συστήματα και το FreeBSD, που συνδυάζει τον πυρήνα της οικογένειας BSD, με μία σειρά εφαρμογών και utilities που φτιάχτηκαν από τρίτους. Τα περιβάλλοντα εργασίας για τελικούς χρήστες που θα μελετήσουμε, θα είναι το KDE και το GNOME, που αποτελούν, μετά βεβαιότητας, τα πλέον διαδεδομένα και δημοφιλή περιβάλλοντα. Για τους διακομιστές, που αποτελούν έναν από τα κυριότερους τομείς των Ελεύθερων συστημάτων, θα εξετάσουμε, σε αυτό το κεφάλαιο, το Apache, που είναι ο παγκόσμιος ηγέτης στην αγορά διακομιστών του Διαδικτύου (*WWW servers*). Κατ' αναλογία, θα παρουσιάσουμε το Mozilla, που είναι ένας από τους διαδικτυακούς πελάτες (*WWW clients*), αν και στην πραγματικότητα είναι πολύ περισσότερο από αυτό και, πάνω στον οποίο, μπορούμε να βασιστούμε, στο πλαίσιο του κόσμου του Ελεύθερου Λογισμικού. Το τελευταίο project που θα αναλύσουμε σε αυτό το κεφάλαιο, θα είναι το OpenOffice.org, μία ελεύθερη Σουίτα εφαρμογών Γραφείου.

Σκεφτήκαμε πως θα ήταν σκόπιμο να μελετήσουμε τις λεπτομέρειες δύο από τις πιο δημοφιλείς διανομές, του Red Hat Linux και του Debian GNU/Linux, και να συγκρίνουμε το μέγεθός τους με άλλα ευρέως χρησιμοποιούμενα συστήματα, σαν τα Windows ή το Solaris. Τέλος, συμπεριλάβαμε και το Eclipse, ένα πολυγλωσσικό περιβάλλον ανάπτυξης λογισμικού.

Μετά τη συζήτηση όλων αυτών των διαφορετικών μελετών περίπτωσης, παραθέτουμε έναν πίνακα που δείχνει τα πιο σημαντικά χαρακτηριστικά της κάθε εφαρμογής ή του κάθε project. Ένα από τα στοιχεία που οι αναγνώστες, ίσως, θεωρήσουν από τα πιο αναπάντεχα, είναι τα αποτελέσματα των εκτιμήσεων του κόστους, της διάρκειας και του αριθμού των developers που απαιτούνται.

Καταλήξαμε σε αυτά τα αποτελέσματα με τη χρήση μεθόδων που τυπικά χρησιμοποιούνται στο πεδίο της Τεχνολογίας του Ελεύθερου Λογισμικού, ειδικά με το Μοντέλο Εκτίμησης Κόστους Λογισμικού COCOMO. Το μοντέλο COCOMO (*Software Engineering Economics*, 1981) [93] λαμβάνει υπ' όψη, ως αρχική μέτρηση, τον αριθμό γραμμών πηγαίου κώδικα και, κατόπιν,

συντάσσει μία εκτίμηση του συνολικού κόστους, του χρόνου ανάπτυξης και της προσπάθειας που απαιτούνται για τη δημιουργία του λογισμικού. Το μοντέλο COCOMO σχεδιάστηκε για τις “κλασικές” διαδικασίες δημιουργίας λογισμικού (μοντέλα ανάπτυξης τύπου καταρράκτη ή τύπου V), είτε για μεσαίου μεγέθους, είτε και για μεγάλου μεγέθους projects: συνεπώς, οι αριθμοί στους οποίους θα καταλήξει, μετά την εφαρμογή του σε κάποιες από τις περιπτώσεις που θα μελετήσουμε, θα πρέπει να ερμηνευθούν με κάποιες επιφυλάξεις. Σε κάθε περίπτωση, τα αποτελέσματα αυτά θα μας βοηθήσουν να πάρουμε μία ιδέα σχετικά με αυτή καθ' εαυτή την κλίμακα πάνω στην οποία δουλεύουμε, καθώς και για το μέγεθος της ισχυρής προσπάθειας που θα είχε απαιτηθεί για την επίτευξη των ίδιων αποτελεσμάτων με το μοντέλο ανάπτυξης του ιδιοταγούς λογισμικού.

Σε γενικές γραμμές, από όλους τους αριθμούς που προκύπτουν με το μοντέλο COCOMO, εκείνοι που εντυπωσιάζουν περισσότερο είναι οι εκτιμήσεις περί κόστους. Σε αυτή την εκτίμηση, λαμβάνονται υπ' όψη δύο παράγοντες: ο μέσος μισθός ενός developer και τα γενικά έξοδα. Για τον υπολογισμό του εκτιμώμενου κόστους, παίρνουμε το επίπεδο μέσου μισθού ενός προγραμματιστή συστημάτων, ολικής απασχόλησης (*full-time*), που προκύπτει από την έκδοση του έτους 2000 “Salary survey 2000” [235]. Τα γενικά έξοδα είναι οι επιπλέον δαπάνες που όλες οι εταιρείες πρέπει να καταβάλλουν για να μπορέσει να κυκλοφορήσει το προϊόν, ανεξαρτήτως από τους μισθούς που πληρώνουν στους προγραμματιστές. Αυτά περιλαμβάνουν τους μισθούς για τις γραμματείες και την ομάδα marketing, μέχρι και το κόστος των φωτοτυπιών, του φωτισμού, του εξοπλισμού, κλπ. Συμπερασματικά, το κόστος που υπολογίζεται με το μοντέλο COCOMO είναι το συνολικό κόστος που θα έχει μία εταιρεία, για να μπορέσει να κατασκευάσει ένα λογισμικό των συγκεκριμένων διαστάσεων και δεν θα πρέπει να ξεχνάμε ότι μόνον ένα μέρος αυτού του ποσού θα πάει στους προγραμματιστές, για τον σχεδιασμό του software. Όταν ληφθεί και αυτό υπ' όψιν, τότε, συνολικά, τα κόστη δεν φαίνονται και τόσο υπερβολικά.

9.1. Ο πυρήνας Linux

Ο πυρήνας Linux αποτελεί, χωρίς αμφιβολία, την κορυφαία εφαρμογή του Ελεύθερου Λογισμικού, σε τέτοιο βαθμό που, ενώ δεν είναι παρά ένα μικρό μέρος του όλου συστήματος, το όνομά του χρησιμοποιείται για να το ορίσει συνολικά. Επιπλέον, θα μπορούσε κανείς να πει ότι το Ελεύθερο Λογισμικό αυτό καθ' εαυτό, συγχέεται συχνά με το Linux, πράγμα που αποτελεί μεγάλο σφάλμα, αφού υπάρχουν ελεύθερα λογισμικά που τρέχουν σε συστήματα που δεν βασίζονται στο Linux (πράγματι, ένας από τους μεγαλύτερους στόχους του κινήματος, αλλά και πολλών projects Ελεύθερου Λογισμικού, είναι η δημιουργία εφαρμογών που να μπορούν να τρέχουν σε πολλά και διαφορετικά περιβάλλοντα). Από την άλλη μεριά, υπάρχουν επίσης και εφαρμογές που δουλεύουν σε Linux και που δεν είναι πραγματικά Ελεύθερο Λογισμικό (σαν το Acrobat Reader, το ιδιοταγές πρόγραμμα ανάγνωσης κειμένων PDF, για το οποίο υπάρχει και μία έκδοση για Linux).

Σημείωση

Υπάρχουν, στις μέρες μας, διάφορα projects που ενσωματώνουν και διανέμουν ελεύθερες εφαρμογές που τρέχουν σε συστήματα Windows, για να μην καταλήξει το Ελεύθερο Λογισμικό να συνδέεται μόνο με τα συστήματα Linux. Ένας από τους πρωτοπόρους σε αυτό τον τομέα (που κατέστη, ίσως, και ο πιο γνωστός και ολοκληρωμένος) ήταν το GNUWin, που διενέμετο σε CDs αυτόματης εκκίνησης (self-bootable), που περιείχαν περισσότερες από 100 ελεύθερες εφαρμογές για συστήματα Win32. Οι περισσότερες από αυτές τις εφαρμογές κυκλοφορούν επίσης μέσα στις διάφορες κοινές διανομές GNU/Linux, πράγμα που κατέστησε το GNUWin ένα καλό εργαλείο για να οδηγήσει προς μία βαθμιαία και εύκολη μετάβαση από ένα σύστημα Windows σε ένα σύστημα GNU/Linux. Από τις αρχές του 2007, υπάρχουν και άλλα παρόμοια συστήματα που διατίθενται, σαν το WinLibre.

9.1.1. Ιστορία του Linux

Η ιστορία του Linux είναι μία από τις πιο γνωστές ιστορίες στον κόσμο του Ελεύθερου Λογισμικού, μάλλον διότι έχει όλα τα χαρακτηριστικά ενός μύθου, παρά της ιστορίας ενός προγράμματος για υπολογιστές. Το 1991, ένας Φινλανδός φοιτητής που λεγόταν Linux Torvalds, αποφάσισε ότι ήθελε να μελετήσει πως να χρησιμοποιεί protected mode 386 σε ένα μηχάνημα που τα περιορισμένα του οικονομικά του επέτρεπαν να αγοράσει. Εκείνο τον καιρό, υπήρχε ένας πυρήνας στο λειτουργικό σύστημα που λεγόταν Minix, που είχε σχεδιασθεί για ακαδημαϊκούς σκοπούς, καθώς και για χρήση σε πανεπιστημιακά μαθήματα για τα Λειτουργικά Συστήματα. Αυτός ο πυρήνας χρησιμοποιείται μέχρι και σήμερα. Ο Andrew Tanenbaum, ένας από τους Καθηγητές Πανεπιστημίου με το μεγαλύτερο κύρος, ήταν ο επικεφαλής μίας ομάδας που εργαζόταν για την ανάπτυξη του Minix, βασίζόμενος σε παραδοσιακά συστήματα Unix. Το Minix ήταν ένα περιορισμένο σύστημα, αλλά πολύ ικανό και καλοσχεδιασμένο, και αποτελούσε το κέντρο μίας μεγάλης ακαδημαϊκής αλλά και τεχνικής κοινότητας.

Το Minix κυκλοφορούσε υπό μία Ελεύθερη Άδεια κυκλοφορίας και μπορούσε να χρησιμοποιείται για ακαδημαϊκούς σκοπούς, αλλά είχε το μεγάλο μειονέκτημα ότι οι άνθρωποι που δεν εργάζονταν ή δεν σπούδαζαν στο Πανεπιστήμιο του Amsterdam, δεν μπορούσαν να συνεισφέρουν βελτιώσεις σε αυτό. Αντιθέτως, αυτές οι βελτιώσεις έπρεπε να γίνουν ξεχωριστά, συνήθως με την χρήση patches. Αυτό σήμαινε ότι στην πράξη, υπήρχε μία επίσημη έκδοση του Minix που χρησιμοποιούσε ο καθένας και, από κει και πέρα, μία μεγάλη ποικιλία patches που έπρεπε να ενσωματωθούν αργότερα, για να προσδώσουν επιπρόσθετες λειτουργίες.

Στα μισά του 1991, ο Linus, τότε ένας άγνωστος Φινλανδός φοιτητής, έστειλε ένα μήνυμα στο newsgroup του Minix, αναγγέλλοντας ότι επρόκειτο να αρχίσει να δουλεύει πάνω σε έναν πυρήνα λειτουργικού συστήματος που θα βασιζόταν στο Minix, ξεκινώντας από μηδενική βάση, γράφοντας κώδικα. Εκείνο τον καιρό, αν και ο Linus δεν είχε πει με σαφήνεια ότι επρόκειτο να τον κυκλοφορήσει με μία Άδεια Ελεύθερου Λογισμικού, σημείωσε ότι το σύστημα που επρόκειτο να δημιουργήσει δεν θα είχε τους *φραγμούς* που είχε το Minix. Αυτό που ήταν, όμως, άγνωστο στον ίδιο και, ίσως, δεν είχε ούτε καν αυτή την πρόθεση, ήταν ότι έκανε το πρώτο βήμα προσέλκυσης όλης της κοινότητας, που μέχρι τότε επικεντρωνόταν γύρω από το Minix, για να την κάνει δική του.

Η έκδοση 0.02, που ανάγεται στον Οκτώβριο του 1991, παρά το ότι ήταν πολύ περιορισμένη, μπορούσε ήδη να τρέξει τερματικά τύπου bash και τον μεταγλωττιστή GCC. Κατά τους επόμενους μήνες, ο αριθμός των εξωτερικών συνεισφορών μεγάλωσε σε τέτοιο σημείο που τον Μάιο του 1992, ο Linus ήταν σε θέση να δημοσιεύσει την έκδοση 0.95, που αναγνωρίστηκε δημόσια ως σχεδόν σταθερή. Εν τούτοις, απέμεναν ακόμη πολλά να γίνουν, πριν την έκδοση 1.0, η οποία, συνήθως, θεωρείται ως η πρώτη σταθερή έκδοση. Τον Δεκέμβριο του 1993, για παράδειγμα, δόθηκε στην κυκλοφορία η έκδοση 0.99pl14 (που αποτελούσε την 14η διορθωμένη έκδοση της αρχικής version 0.99). Τον Μάρτιο του 1994, γεννήθηκε τελικά το Linux 1.0. Εκείνη την εποχή πλέον, το Linux κυκλοφορούσε υπό τους όρους της Άδειας GPL. Σύμφωνα με τον ίδιο τον Torvalds, αυτή ήταν μία από τις καλύτερες αποφάσεις που είχε ποτέ λάβει, καθώς στάθηκε μεγάλη βοήθεια στη διάδοση και προσέλκυση δημοτικότητας για τον πυρήνα του. Στο *"Evolution in open source software: a case study"*, [128] υπάρχει μία εμβριθής ανάλυση της εξέλιξης των διαφόρων εκδόσεων του πυρήνα Linux, εστιάζοντας στις δυνατότητες εξέλιξης και την τμηματικότητα (*modularity*).

Σημείωση

Ακόμη ένα σημαντικό γεγονός στα αρχεία του Ελεύθερου Λογισμικού υπήρξε η αντιπαράθεση που έλαβε χώρα περί τα τέλη του Ιανουαρίου του 1992, στο Minix newsgroup, ανάμεσα στον Andrew Tanenbaum και τον Linus Torvalds. Ο Tanenbaum, ο οποίος ήταν πιθανώς ενοχλημένος από την επιτυχία του Torvalds με το "παιγνίδάκι του", επετέθη στο

Linux και στον Linus, με έναν μάλλον δυσανάλογο τρόπο. Το βασικό του επιχείρημα ήταν ότι το Linux ήταν ένα *μονολιθικό* σύστημα (ο πυρήνας ενσωματώνει όλους τους χειριστές/ *handlers* του συστήματος με όλα τα υπόλοιπα τμήματα) και όχι ένα σύστημα *μικροπυρήνα/ microkernel* (όπου ο πυρήνας έχει ένα σχεδιασμό κατά ενότητες/*modular design*, που σημαίνει ότι μπορεί να είναι κατά πολύ μικρότερος και ότι τα αρθρώματα (*modules*) μπορούν να φορτώνονται όποτε χρειάζονται). Μπορείτε να διαβάσετε την αρχική αντιπαράθεση, ακριβώς όπως έλαβε χώρα, στο "*The Tanenbaum-Torvalds debate*" newsgroup [214].

9.1.2. Ο τρόπος λειτουργίας του Linux

Ο τρόπος με τον οποίο εργαζόταν ο Torvalds δεν ήταν και πολύ κοινός τον καιρό εκείνο⁽⁹⁾. Η ανάπτυξη κώδικα βασιζόταν κυρίως σε μία λίστα ταχυδρομείου (*mailing list*). Η λίστα ταχυδρομείου ήταν ένα μέρος όπου όχι μόνο οι άνθρωποι διαπληκτιζόταν, αλλά και στο οποίο ελάμβαναν χώρα οι εξελίξεις. Αυτό συνέβαινε επειδή ο Torvalds ήθελε να αντικατοπτρίζεται ολόκληρη η ζωή του project μέσα σε αυτή τη λίστα. Αυτός ήταν και ο λόγος για τον οποίο ζητούσε από τον κόσμο να στέλνουν τα δικά τους patches προς την λίστα. Αντιθέτως με εκείνο που θα περίμεναν οι περισσότεροι (δηλαδή να στέλνονται τα patches υπό μορφή συνημμένων), ο Linus προτίμησε να υπάρχει ο πηγαίος κώδικας μέσα στο κυρίως σώμα του κειμένου, ούτως ώστε να μπορεί τόσο ο ίδιος, όσο και οι άλλοι, να προβαίνουν σε σχόλια πάνω σε αυτόν. Παρ' όλα αυτά, αν και πολύς κόσμος έδινε τη γνώμη του και έστελνε διορθώσεις ή κάποιες νέες, πρόσθετες λειτουργίες, την τελευταία λέξη την είχε πάντοτε ο ίδιος ο Linus Torvalds, ο οποίος και αποφάσιζε για το ποιος κώδικας θα ενσωματωνόταν στον πυρήνα Linux. Σε μεγάλο βαθμό, αυτός είναι και ο τρόπος λειτουργίας μέχρι και το 2007.

Σημείωση

Η καθιέρωση του Linus Torvalds ως ενός "καλοπροαίρετου δικτάτορα" έδωσε τροφή σε έναν μεγάλο αριθμό ανεκδότων μέσα στο project. Ελέγχθη, για παράδειγμα, ότι αν μία ιδέα αρέσει, τότε θα πρέπει να υλοποιείται. Αν όχι, τότε και πάλι θα πρέπει να υλοποιείται. Το συμπέρασμα, λοιπόν, είναι ότι οι καλές ιδέες δεν έχουν καμία χρησιμότητα (χωρίς τον κώδικα, ασφαλώς). Σε ένα διαφορετικό στυλ, ακόμη και αν μία υλοποίηση δεν καλοαρέσει, είναι βασικό να επιμένει κανείς. Μία πολύ γνωστή περίπτωση είναι εκείνη του Gooch, μπροστά στον οποίο ο Ιώβ δεν ήταν παρά ένας απλός μαθητής. Ο Gooch έφτιαξε πάνω από 146 παράλληλα patches, μέχρι που ο Linus αποφάσισε επιτέλους να τα ενσωματώσει στην επίσημη έκδοση του πυρήνα.

Άλλη μία από τις καινοτόμες ιδέες του Torvalds ήταν να αναπτυχθούν δύο παράλληλοι κλάδοι του πυρήνα: ο σταθερός (με τον δεύτερο αριθμό της έκδοσης να είναι συνήθως ζυγός, πχ. 2.4.18) και ο ασταθής πυρήνας (με τον δεύτερο αριθμό της έκδοσης μονό, πχ. 2.5.12). Ως συνήθως, ο Torvalds είναι πάντοτε το πρόσωπο που αποφασίζει τι ακριβώς θα καταλήξει και σε ποιον από τους δύο κλάδους (οι πιο πολλές από τις αμφιλεγόμενες αποφάσεις σχετίζονται, ακριβώς, με αυτό το σημείο). Σε κάθε περίπτωση, ο πυρήνας Linux δεν προβλέπει προγραμματισμένη διάθεση εκδόσεων σε τακτά χρονικά διαστήματα: θα είναι έτοιμος όταν θα είναι έτοιμος και, στο εν τω μεταξύ, εμείς μπορούμε μόνο να περιμένουμε. Οι περισσότεροι αναγνώστες θα έχουν, σίγουρα, συνειδητοποιήσει έως τώρα ότι την απόφαση για το αν το σύστημα είναι έτοιμο ή όχι, θα την λάβει αποκλειστικά ο Linus.

⁽⁹⁾ Η ηλεκτρονική διεύθυνση της λίστας είναι linux-kernel@vger.kernel.org. Τα ιστορικά μηνύματα μπορείτε να τα διαβάσετε στη διεύθυνση <http://www.uwsg.indiana.edu/hypermail/linux/kernel/>.

Η μέθοδος ανάπτυξης που χρησιμοποιήθηκε στο Linux, αποδείχθηκε πολύ αποτελεσματική από την σκοπιά των αποτελεσμάτων: το Linux είναι πολύ σταθερό και τα οποιαδήποτε bugs διορθώνονται εξαιρετικά γρήγορα (ενίοτε, ακόμη και μέσα σε μερικά λεπτά), καθώς διαθέτει χιλιάδες developers. Με αυτές τις προϋποθέσεις, όταν υπάρχει ένα bug, η πιθανότητα να το εντοπίσει κάποιος είναι πολύ υψηλή, και αν το άτομο που το ανακάλυψε δεν μπορεί να το διορθώσει, τότε κάποιος άλλος θα εμφανιστεί που θα βρει την λύση πολύ γρήγορα. Προς ανακεφαλαίωση, αυτό δείχνει ότι το

Linux μπορεί να υπολογίζει σε χιλιάδες ανθρώπους που εργάζονται πάνω στην ανάπτυξη του κάθε μήνα, οπότε αυτός είναι και ο λόγος για τον οποίο η επιτυχία του δεν προκαλεί, εν τέλει, και ιδιαίτερη κατάπληξη.

Θα πρέπει, πάντως, να σημειωθεί ότι αυτός ο τρόπος δουλειάς είναι πολύ δαπανηρός σε ότι αφορά την κατανάλωση πόρων. Δεν είναι ασύνηθες να υπάρχουν πολλές αμοιβαία ασύμβατες προτάσεις για μία νέα λειτουργία ή να αποστέλλονται μία ντουζίνα patches για το ίδιο bug. Στις περισσότερες περιπτώσεις, μόνον ένα από όλα αυτά τα patches θα ενσωματωθεί τελικά στον πυρήνα, πράγμα που σημαίνει ότι όλη η υπόλοιπη ενέργεια και ο χρόνος που καταναλώθηκαν για τα άλλα patches, εκ μέρους των άλλων developers, θα έχουν πάει όλα χαμένα εις μάτην. Το αναπτυξιακό μοντέλο του Linux είναι, συνεπώς, ένα μοντέλο που μπορεί να δουλεύει πολύ καλά για το Linux, αλλά που δεν μπορούν όλα τα projects να έχουν την πολυτέλεια να το εφαρμόζουν.

9.1.3. Η παρούσα κατάσταση του Linux

Στις αρχές του 2007, ο πυρήνας Linux βρισκόταν στην έκδοση 2.6, και περιελάμβανε, από την πλευρά των βελτιώσεων που είχαν γίνει στην έκδοση 2.4, NUMA (*Non-Uniform Memory Access*/Μη-Ομοιόμορφη πρόσβαση Μνήμης, σε μεγάλη χρήση στους πολλαπλούς επεξεργαστές), νέο σύστημα αρχείων, βελτιώσεις στον τομέα της επικοινωνίας των ασυρμάτων δικτύων και στην επεξεργασία ήχου (*ALSA*), καθώς και πολυάριθμες άλλες βελτιώσεις (αν ενδιαφέρεστε για τις λεπτομέρειες των αλλαγών, σε σύγκριση με τις προηγούμενες εκδόσεις, μπορείτε να ανατρέξετε στο "The wonderful world of Linux 2.6" ("*Ο υπέροχος κόσμος του Linux 2.6*") [186]).

Το μοντέλο ανάπτυξης του Linux υπέστη μερικές αλλαγές στα τελευταία χρόνια. Αν και οι λίστες ταχυδρομείου για την ανάπτυξη αποτελούν ακόμη την ψυχή του συστήματος, ο κώδικας δεν χρειάζεται να περάσει απαραίτητα μέσα από τη λίστα. Ένα από τα πράγματα που συνεισέφεραν σε αυτή την αλλαγή σε μεγάλο βαθμό ήταν το BitKeeper, ένα ιδιοταγές σύστημα που εκτελεί τον έλεγχο των αναθεωρήσεων κώδικα (*revision control*), και το οποίο είχε αναπτυχθεί από την εταιρεία BitMover, ακολουθώντας αυστηρά τις συστάσεις του Linus Torvalds. Η χρήση αυτού του ιδιοταγούς εργαλείου οδήγησε σε μία μεγάλη διαμάχη, στην οποία ο Linus, συνεπής στο στυλ του, απέδειξε και πάλι τον πραγματισμό του, καθώς τόσο για τον ίδιο, όσο και για πολλούς άλλους, το σύστημα CVS για τον έλεγχο των εκδόσεων ήταν πλέον απαρχαιωμένο. Οι διαφωνίες κάποια στιγμή κόπασαν, με την ανάπτυξη του git, ενός συστήματος για τον έλεγχο των αναθεωρήσεων με χαρακτηριστικά παρόμοια με εκείνα που διέθετε το BitKeeper, που χρησιμοποιείτο, μέχρι τότε, για την ανάπτυξη του Linux. Πιο συγκεκριμένα, η διαδικασία ανάπτυξης του Linux ακολουθεί μία πυραμιδική ιεραρχία, στην οποία οι developers προτείνουν διάφορα patches, που διανέμονται μεταξύ των διαφόρων επιπέδων μέσω e-mail, και που θα πρέπει να γίνουν αποδεκτά από το παραπάνω επίπεδο της κλίμακος, το οποίο αποτελείται από ελεγκτές και συντηρητές αρχείων. Οι συντηρητές υποσυστήματος βρίσκονται σε μία ανώτερη βαθμίδα, ενώ ο Linus Torvalds και ο Andrew Morton βρίσκονται στο κορυφαίο επίπεδο και έχουν την τελευταία λέξη όταν πρόκειται για την αποδοχή των εν λόγω patches.

Για να ανακεφαλαιώσουμε, ο παρακάτω πίνακας παρέχει ένα είδος ακτινογραφίας του Linux project, δείχνοντας πως τώρα διαθέτει παραπάνω από 5.000.000 γραμμές κώδικα και ότι, άρα, μπορεί να συμπεριληφθεί ανάμεσα στα μεγαλύτερα projects Ελεύθερου λογισμικού (μαζί με το Mozilla και το OpenOffice.org). Σε ότι αφορά τις εκτιμήσεις για το πόσος χρόνος θα χρειαζόταν για τον σχεδιασμό ενός τέτοιου project και τον μέσο αριθμό developers που θα ήταν απαραίτητοι, θα πρέπει να σημειώσουμε ότι ο χρόνος είναι σίγουρα πολύ λιγότερος από όσο διάστημα υπάρχει το ίδιο το Linux. Από την άλλη μεριά, αυτό αντισταθμίζεται άνετα από την τελευταία λεπτομέρεια,

δεδομένου ότι ο μέσος αριθμός developers που εργάζονται υπό καθεστώς πλήρους απασχόλησης, που θα ήταν απαραίτητοι για ένα παρόμοιο project, είναι πολύ υψηλότερος από τον αριθμό που εκείνων που απασχολήθηκαν ποτέ στο Linux.

Σημείωση

Η εκτίμηση κόστους που δείχνει το μοντέλο COCOMO, είναι της τάξεως των 215 εκατομμ. δολαρίων ΗΠΑ: ένα ποσό που, αν το θέταμε στα πλαίσια των καθημερινών ποσών που θα μπορούσε κανείς να σκεφθεί, θα ήταν διπλάσιο του ποσού που θα μπορούσαν να πληρώσουν ακόμη και οι καλύτεροι ποδοσφαιρικοί σύλλογοι για έναν σπουδαίο ποδοσφαιριστή.

Πίνακας 4. Ανάλυση του Linux

Ιστοσελίδα	http://www.kernel.org
Αρχή του project	1 ^ο μήνυμα στο news.comp.os.minix: Αύγ. 1991
Άδεια	GNU GPL
Έκδοση που αναλύθηκε	2.6.20 (σταθερή έκδοση στις 20/02/2007)
Γραμμές Πηγαίου Κώδικα.	5,195,239
Εκτίμηση κόστους (σύμφωνα με το βασικό COCOMO)	\$ 215,291,772
Εκτίμηση του χρονοδιαγράμματος για τον σχεδιασμό (σύμφωνα με το βασικό COCOMO)	8.83 έτη (105.91 μήνες)
Εκτίμηση του μέσου αριθμού developers (σύμφωνα με το βασικό COCOMO)	180.57
Κατά προσέγγιση Αριθμός των developers	Εκτιμώνται σε χιλιάδες (αν και στα credits εμφανίζονται μόνο κατά εκατοντάδες[219])
Εργαλεία για την υποστήριξη της ανάπτυξης	Η Ταχυδρομική Λίστα (Mailing list) και το git

Η σύνθεση του Linux, από τη σκοπιά των γλωσσών προγραμματισμού, δείχνει ότι υπάρχει μία ξεκάθαρη κυριαρχία της C, η οποία θεωρείται ιδεώδης γλώσσα για τη σχεδίαση συστημάτων που έχουν την ταχύτητα ως κρίσιμη παράμετρο. Όταν η ταχύτητα αποτελεί μία τόσο βασική απαίτηση που δεν μπορεί να καλύψει ούτε η C, τότε χρησιμοποιείται ευθέως μία γλώσσα assembly για τον προγραμματισμό και αυτό, όπως θα δούμε, συμβαίνει αρκετά συχνά. Το μειονέκτημα αυτής της γλώσσας assembly, σε σύγκριση με την C, είναι ότι δεν είναι φορητή στον ίδιο βαθμό. Η κάθε αρχιτεκτονική διαθέτει το δικό της σύνολο ιδιαίτερων εντολών, πράγμα που σημαίνει ότι πολύς κώδικας που γράφτηκε για μία αρχιτεκτονική σε γλώσσα assembly, θα πρέπει να μεταφρασθεί και στις άλλες αρχιτεκτονικές.

Η συχνότητα χρήσης των άλλων γλωσσών, όπως φαίνεται από τον πίνακα που παραθέτουμε, είναι εντελώς περιθωριακή: περιορίζονται σε εγκαταστάσεις, σε συγκεκριμένες λειτουργίες και σε κάποια βοηθήματα για την ανάπτυξη. Η έκδοση που αναλύσαμε για το βιβλίο αυτό, ήταν το Linux 2.6.20, όπως δημοσιεύθηκε στις 20/02/2007 (χωρίς να συμπεριλαμβάνει οποιαδήποτε από τα μεταγενέστερα patches).

Πίνακας 5. Γλώσσες προγραμματισμού που χρησιμοποιούνται στο Linux

Γλώσσα προγραμματισμού	Γραμμές Κώδικα	Ποσοστό
C	4,972,172	95.71%
Perl	210,693	4.06%
Yacc	3,224	0.06%
Shell	2,632	0.05%
Assembler	2,203	0.04%

9.2. To FreeBSD

Όπως αναφέραμε ήδη στο κεφάλαιο για την ιστορία του Ελεύθερου Λογισμικού, πέραν του δημοφιλούς GNU/ Linux, υπάρχουν και άλλα είδη ελεύθερων Λειτουργικών Συστημάτων. Μία από αυτές τις οικογένειες συστημάτων είναι και οι "κληρονόμοι" των διανομών του Πανεπιστημίου του Berkeley, στην California (ΗΠΑ): τα συστήματα τύπου BSD. Το παλιότερο και πιο γνωστό σύστημα BSD είναι το FreeBSD, το οποίο δημιουργήθηκε στις αρχές του 1993, όταν ο Bill Jolitz σταμάτησε την δημοσίευση των επίσημων ενημερώσεων του 386BSD. Με την βοήθεια της εταιρείας Walnut Creek CDROM, που αργότερα άλλαξε το όνομά της σε BSDi, μία ομάδα εθελοντών αποφάσισε να αρχίσει τη δημιουργία αυτού του ελεύθερου συστήματος.

Ο βασικός στόχος του project FreeBSD ήταν η δημιουργία ενός λειτουργικού συστήματος που να μπορεί να χρησιμοποιηθεί χωρίς κανενός είδους δεσμεύσεις ή εξαρτήσεις, αλλά να έχει και όλα τα πλεονεκτήματα της ελεύθερης διαθεσιμότητας του κώδικα καθώς και να είναι επεξεργασμένο με τόση προσοχή ώστε να μπορεί να εγγυηθεί την ποιότητα του προϊόντος. Ο χρήστης έχει την ελευθερία να κάνει οτιδήποτε του αρέσει με το λογισμικό, είτε τροποποιώντας το σύμφωνα με τις επιθυμίες του, είτε αναδιανέμοντας το υπό μία ανοικτή μορφή, ή ακόμη και με κλειστή μορφή, υπό τους όρους που ο ίδιος ο χρήστης επιθυμεί, με ή χωρίς τροποποιήσεις. Όπως δείχνει και το όνομα από μόνο του, το FreeBSD project βασίζεται στη φιλοσοφία των Αδειών BSD licenses.

9.2.1. Η ιστορία του FreeBSD

Η έκδοση 1.0 εμφανίστηκε κατά το τέλος του 1993 και βασιζόταν στο 4.3BSD Net/2 και στο 386BSD. 4.3BSD Net/2, διέθετε κώδικα που είχε γραφτεί κατά τη δεκαετία του 1970, όταν το Unix αναπτυσσόταν από την AT&T, πράγμα που, όπως αποδείχθηκε, οδήγησε σε μία σειρά νομικών προβλημάτων που δεν είχαν επιλυθεί μέχρι το 1995, όταν δημοσιεύθηκε το FreeBSD 2.0, χωρίς, όμως, τον αρχικό κώδικα που είχε αναπτυχθεί από την AT&T. Τώρα βασιζόταν στο 4.4BSD- Lite, μία ελαφρύτερη έκδοση του 4.4BSD (όπου πολλά από τα αρθρώματα/ *modules* είχαν διαγραφεί για νομικούς λόγους, πέραν του γεγονότος ότι η μεταφορά στα συστήματα Intel ήταν ακόμη ατελής) και που δόθηκε στην κυκλοφορία από το Πανεπιστήμιο της Καλιφόρνιας.

Η ιστορία του FreeBSD δεν θα ήταν πλήρης αν παραλείπαμε να αναφέρουμε τις "αδελφικές" διανομές της, το NetBSD και το OpenBSD. Το NetBSD εμφανίστηκε σαν έκδοση 0.8, περί τα μέσα του 1993. Ο βασικός στόχος ήταν να μπορεί να είναι ιδιαίτερα φορητό (αν και αρχικά ήταν μόνο μία προσαρμογή για το i386). Κατά συνέπεια, το σλόγκαν των προϊόντων ήταν: "*Ασφαλώς και τρέχει NetBSD*". Το OpenBSD γεννήθηκε μετά την διάσπαση του NetBSD που προεκλήθη λόγω

φιλοσοφικών διαφορών (αλλά και προσωπικών διαφορών) μεταξύ των developers, κατά τα μέσα του 1996. Η εστίαση είναι κυρίως στην ασφάλεια και την κρυπτογράφηση και λένε ότι είναι το ασφαλέστερο λειτουργικό που υπάρχει, ενώ, χάρις στο ότι βασίζεται στο NetBSD, είναι επίσης και εξαιρετικά φορητό.

9.2.2. Η διαδικασία ανάπτυξης στο FreeBSD

Το μοντέλο ανάπτυξης που χρησιμοποιεί το FreeBSD project βασίζεται κυρίως σε δύο εργαλεία: Το σύστημα ελέγχου εκδόσεων CVS (version control system) και το λογισμικό GNATS, για τον εντοπισμό των bug. Το όλο project βασίζεται σε αυτά τα δύο εργαλεία, όπως προκύπτει και από το γεγονός ότι δημιουργήθηκε μία οργανωτική ιεραρχία βασισμένη σε αυτά τα εργαλεία. Πράγματι, εκείνοι που διαθέτουν το μεγαλύτερο κύρος στο project είναι οι *committers* (δηλ. οι developers με δικαίωμα write-access στο αποθετήριο του CVS), είτε άμεσα είτε έμμεσα, και οι οποίοι επιλέγονται από την *κεντρική ομάδα* (δηλ. το Διευθυντικό Συμβούλιο), όπως θα δούμε στο επόμενο κεφάλαιο.

Δεν χρειάζεται να είστε απαραίτητα *committer* για να μπορείτε να υποβάλετε αναφορές σχετικά με bugs στο GNATS, πράγμα που σημαίνει ότι όποιος θέλει μπορεί να το κάνει. Όλες αυτές οι (ανοιχτές) συνεισφορές στο GNATS κρίνονται από έναν *committer*, που μπορεί να αναθέσει την (αναλυθείσα) εργασία σε έναν άλλον *committer*, ή να ζητήσει περισσότερες πληροφορίες από εκείνον που υπέβαλε αρχικά το bug report. Υπάρχουν καταστάσεις στις οποίες το bug επιλύθηκε για κάποιους πρόσφατους κλάδους του συστήματος, οπότε θα διευκρινίζεται με την ένδειξη “suspended”. Σε κάθε περίπτωση, άπαξ και διορθωθεί το σφάλμα, ο στόχος είναι να κλείνει η αναφορά.

Το FreeBSD διανέμει το λογισμικό του σε δύο μορφές: από τη μία μεριά, τα *ports*, ένα σύστημα που κατεβάζει τους πηγαίους κώδικες, τους μεταγλωττίζει και εγκαθιστά την εφαρμογή στον τοπικό υπολογιστή και, από την άλλη, τα *πακέτα*, που είναι, απλούστατα, οι πηγαίοι κώδικες των προμεταγλωττισμένων *ports*. Το σπουδαιότερο πλεονέκτημα των *ports*, σε σύγκριση με τα *πακέτα*, είναι ότι επιτρέπουν στον χρήστη να ρυθμίσει, να προσαρμόσει και να βελτιστοποιήσει το software για τον δικό του προσωπικό υπολογιστή. Από την άλλη πλευρά, όταν κάνουμε χρήση του συστήματος των *πακέτων*, καθώς αυτά είναι ήδη προμεταγλωττισμένα, παίρνει πολύ λιγότερο χρόνο για την εγκατάσταση του λογισμικού.

9.2.3. Η διαδικασία λήψης αποφάσεων στο FreeBSD

Το Διευθυντικό Συμβούλιο του FreeBSD, που ως γνωστόν αποκαλείται “κεντρική ομάδα” (core team), έχει την ευθύνη να καθορίζει την κατεύθυνση του project, να εξασφαλίζει ότι επιτυγχάνονται οι στόχοι, αλλά και να μεσολαβεί στις περιπτώσεις διενέξεων μεταξύ των *committers*.

Μέχρι τον Οκτώβριο του 2000, ήταν μία κλειστή ομάδα, όπου μπορούσε κανείς να έχει πρόσβαση μόνο μετά από σαφή πρόσκληση, εκ μέρους της ίδιας της “κεντρικής ομάδας”. Από τον Οκτώβριο του 2000, τα μέλη εκλέγονται περιοδικά και με δημοκρατικό τρόπο από τους *committers*. Οι βασικότεροι κανόνες κατά τις εκλογές της Κεντρικής Ομάδας είναι οι εξής:

- 1) Οι *committers* που συνεισέφεραν τουλάχιστον ένα *commit* μέσα στον τελευταίο χρόνο, έχουν δικαίωμα ψήφου.
- 2) Το Διευθυντικό Συμβούλιο θα ανανεώνεται κάθε δύο χρόνια.

- 3) Τα μέλη του Διευθυντικού Συμβουλίου μπορούν να "καθαιρούνται" με τις ψήφους των 2/3 των *committers*.
- 4) Αν ο αριθμός των μελών του Διευθυντικού Συμβουλίου είναι μικρότερος από επτά, τότε θα πρέπει να προκηρυχθούν νέες εκλογές.
- 5) Προκηρύσσονται νέες εκλογές όταν το 1/3 των *committers* ψηφίζουν προς αυτή την κατεύθυνση.
- 6) Οποιοσδήποτε αλλαγές των κανόνων απαιτούν μία πλειοψηφία των 2/3 των *committers*.

9.2.4. Εταιρείες που εργάζονται γύρω από το FreeBSD

Υπάρχουν πολυάριθμες εταιρείες που προσφέρουν υπηρεσίες και προϊόντα που βασίζονται στο FreeBSD. Το ίδιο το FreeBSD τις απαριθμεί στον ιστότοπο του project. Στην παρούσα ανάλυση του FreeBSD θα μάθουμε παραπάνω για τις πιο σημαντικές πλευρές: το BSDi και το Walnut Creek CDROM.

Το FreeBSD γεννήθηκε, εν μέρει, μετά την ίδρυση της εταιρείας BSDi, το 1991, από τους ανθρώπους της CSRG (Computer Systems Research Group) του Πανεπιστημίου του Berkeley, και θα παρείχε εμπορική υποστήριξη προς το νέο λειτουργικό σύστημα. Εκτός από την εμπορική έκδοση του λειτουργικού συστήματος FreeBSD, η BSDi ανέπτυξε επίσης και άλλα προϊόντα, όπως πχ. ένα διακομιστή Internet και ένα διακομιστή gateway.

Το Walnut Creek CDROM δημιουργήθηκε με στόχο την εμπορικοποίηση του FreeBSD ως τελικού προϊόντος, με τρόπο που να μπορεί να θεωρείται ως μία διανομή, στο στυλ εκείνων που υπάρχουν για το GNU/Linux, αλλά με το FreeBSD. Τον Νοέμβριο του 1998, το Walnut Creek διεύρυνε τους ορίζοντές του με την δημιουργία της πύλης FreeBSD Mall, που θα προωθούσε εμπορικά όλα τα είδη προϊόντων που βασιζόταν στο FreeBSD (αρχίζοντας με αυτή καθ' εαυτή την διανομή, μέχρι και μπλουζάκια t-shirts, περιοδικά, βιβλία, κλπ.), θα φρόντιζε την αναγγελία προϊόντων εκ μέρους τρίτων κατασκευαστών σε αυτή την ιστοσελίδα, καθώς και θα παρείχε επαγγελματική υποστήριξη για το FreeBSD.

Τον Μάρτιο του 2000, το BSDi και το Walnut Creek συγχωνεύθηκαν υπό το όνομα BSDi, για να εργασθούν από κοινού ενάντια στο φαινόμενο Linux, το οποίο υπερσκίαζε ξεκάθαρα τα συστήματα BSD γενικά, αλλά και ειδικότερα το FreeBSD. Ένα χρόνο μετά, τον Μάιο του 2001, η Wind River εξαγόρασε εκείνο το τμήμα που ήταν υπεύθυνο για την δημιουργία του λογισμικού BSDi, με τη σαφή πρόθεση της ενίσχυσης της ανάπτυξης του FreeBSD, για τη χρήση του σε ενσωματωμένα (embedded) και έξυπνα συστήματα, συνδεδεμένα με το Διαδίκτυο.

9.2.5. Η παρούσα κατάσταση του FreeBSD

Σύμφωνα με τα πιο πρόσφατα στοιχεία των δημοσκοπήσεων που διεξάγει ανά τακτά διαστήματα η Netcraft, ο αριθμός των διακομιστών διαδικτύου που τρέχουν FreeBSD είναι περίπου 2 εκατομμύρια. Ένας νέος χρήστης που θα ήθελε να εγκαταστήσει το FreeBSD, θα μπορούσε να επιλέξει ανάμεσα στην έκδοση 6.2 (που θα μπορούσε να θεωρηθεί ως η "σταθερή" έκδοση), ή την πιο προχωρημένη "πilotική έκδοση ανάπτυξης" (development branch). Ενώ η πρώτη προσφέρει μεγαλύτερη σταθερότητα, ιδιαίτερα σε θέματα όπως το συμμετρικό multiprocessing, που υπέστη πλήρη εκ βάθρων ανάπτυξη στις νεότερες εκδόσεις, η δεύτερη επιτρέπει στους χρήστες να

απολαμβάνουν τις πιο πρόσφατες καινοτομίες. Είναι, επίσης, σημαντικό να θυμίσουμε ότι οι πιλοτικές εκδόσεις περιλαμβάνουν και δοκιμαστικό κώδικα (test code), πράγμα που επηρεάζει ελαφρώς την ταχύτητα του συστήματος.

Ένα από τα κορυφαία χαρακτηριστικά του FreeBSD είναι εκείνο που έγινε γνωστό ως οι “φυλακές” (*jails*). Οι φυλακές ελαχιστοποιούν την ζημιά που θα μπορούσε να προκληθεί από μία επίθεση σε βασικές υπηρεσίες δικτύου, σαν το Sendmail για την διακίνηση των emails, ή το BIND (Berkeley Internet Name Domain) για τη διαχείριση ονομάτων. Οι διάφορες υπηρεσίες τοποθετούνται σε μία “φυλακή”, έτσι ώστε να τρέχουν μέσα σε ένα απομονωμένο περιβάλλον. Μπορεί κανείς να διαχειρισθεί αυτές τις “φυλακές” με τη χρήση διαφόρων utilities που περιέχει το FreeBSD.

9.2.6. Ακτινογραφία του FreeBSD

Όπως είπαμε στο προηγούμενο κεφάλαιο, οι λειτουργίες του FreeBSD δεν περιορίζονται μόνο στην ανάπτυξη ενός πυρήνα λειτουργικού συστήματος, αλλά περιλαμβάνουν επίσης την ενσωμάτωση μίας πλειάδας utilities που διανέμονται μαζί, ανάλογα με εκείνο που γίνεται και στις διανομές GNU/Linux. Το ότι η διαδικασία ανάπτυξης του FreeBSD είναι στενά συνδεδεμένη με το σύστημα ελέγχου εκδόσεων CVS, σημαίνει ότι μελετώντας το σύστημα αυτό, μπορούμε να πάρουμε μία καλή ιδέα για το πως είναι δομημένο το FreeBSD. Οι αριθμοί που παρατίθενται πιο κάτω, ανταποκρίνονται στην ανάλυση του FreeBSD που έγινε στις 21/08/2003.

Μία από τις πλέον ενδιαφέρουσες πλευρές του FreeBSD είναι ότι οι αριθμοί του πλησιάζουν πολύ με εκείνους του KDE και του GNOME: το μέγεθος του λογισμικού υπερβαίνει άνετα τα 5 εκατομμύρια γραμμές κώδικα, ο αριθμός των συνολικών αρχείων είναι περίπου 250,000, ενώ ο συνολικός αριθμός των *commits* πλησιάζει τα 2 εκατομμύρια. Είναι, πάντως, ενδιαφέρον να παρατηρήσει κανείς ότι η βασική διαφορά ανάμεσα στο GNOME και το KDE, σε σύγκριση με το FreeBSD, είναι η ηλικία του project. Το FreeBSD έφθασε αισίως το δέκατο έτος του και κυκλοφορεί επί σχεδόν διπλάσιο χρονικό διάστημα σε σχέση με τα περιβάλλοντα εργασίας με τα οποία το συγκρίνουμε. Το ότι το μέγεθος είναι παρόμοιο, παρά το γεγονός ότι η περίοδος ανάπτυξης υπήρξε μακρύτερη, οφείλεται εν μέρει στο ότι το FreeBSD δεν προσήλκυσε τόσους πολλούς developers. Υπάρχει ένας κατάλογος με περίπου 400 developers που έχουν δικαίωμα write-access στο CVS (*committers*), ενώ ο αριθμός των developers που απαριθμούνται στο εγχειρίδιο του FreeBSD είναι περίπου χίλιοι (1000).

Αυτός είναι ο λόγος για τον οποίο η καταγεγραμμένη δραστηριότητα του FreeBSD στο CVS είναι κατώτερη από τον μέσο όρο (500 *commits* ανά ημέρα), σε σχέση με εκείνη που κατεγράφη τόσο για το GNOME (900) όσο και για το KDE (1700, περιλαμβάνοντας και τα αυτόματα *commits*).

Θεωρήσαμε ως βασικό σύστημα του FreeBSD όλα εκείνα που βρίσκονται στον φάκελλο `src/src` της ενότητας *root* module του CVS. Η δραστηριότητα που κατεγράφη στο βασικό σύστημα κατά τη διάρκεια των τελευταίων 10 ετών είναι παραπάνω από 500.000 *commits*. Υπάρχουν περισσότερες από 5.000.000 γραμμές κώδικα, αν και θα πρέπει να θυμόμαστε ότι αυτό δεν περιλαμβάνει μόνον τον πυρήνα, αλλά και πολλά άλλα utilities, όπως παιχνίδια. Αν πάρουμε υπ' όψη μας μόνο τον πυρήνα (που βρίσκεται στον υποφάκελλο `sys`), τότε η κλίμακα είναι της τάξεως του 1.5 εκατομμυρίου γραμμών πηγαίου κώδικα, κατά κύριο λόγο γραμμένες σε C.

Παρουσιάζει ενδιαφέρον η διαπίστωση ότι το χρονικό διάστημα που θα ήταν απαραίτητο σύμφωνα με την εκτίμηση του μοντέλου COCOMO, αντιστοιχεί ακριβώς στον πραγματικό χρόνο που χρειάστηκε το FreeBSD project, αν και η εκτίμηση του μέσου αριθμού developers είναι κατά πολύ

ανώτερο από τον πραγματικό αριθμό. Θα πρέπει, επίσης, να τονίσουμε ότι κατά τον τελευταίο χρόνο, μόνον 75 *committers* κατεγράφησαν ως δραστήριοι, ενώ το μοντέλο COCOMO υποθέτει ότι μετά από 10 χρόνια ανάπτυξης, ο αριθμός developers θα έπρεπε να ήτανε 235.

Τέλος, θα πρέπει να θυμόμαστε, όπως εξ' άλλου έχουμε πει, ότι οι κυρίως δραστηριότητες του FreeBSD περιστρέφονται γύρω από το αποθετήριο του CVS και το σύστημα GNATS για τον έλεγχο των bugs.

Πίνακας 6. Ανάλυση του FreeBSD

Ιστότοπος	http://www.FreeBSD.org
Έναρξη του project	1993
Άδεια	of BSD type
Έκδοση που αναλύθηκε	4.8
Γραμμές Πηγαίου Κώδικα	750.000
Γραμμές Πηγαίου Κώδικα (μόνον πυρήνας)	1.500.000
Αριθμός αρχείων	250.000
Εκτίμηση κόστους	\$ 325.000.000
Εκτίμηση απαιτούμενου χρόνου ως την ολοκλήρωση	10.5 χρόνια (126 μήνες)
Εκτίμηση μέσου αριθμού developers	235
Κατά προσέγγιση αριθμός developers	400 <i>committers</i> (1.000 συνεργάτες)
Αριθμός δραστήριων <i>committers</i> κατά το τελευταίο έτος	75 (λιγότερο από το 20% του συνόλου)
Αριθμός δραστήριων <i>committers</i> κατά τα τελευταία 2 χρόνια	165 (περίπου το 40% του συνολικού)
Αριθμός των <i>commits</i> στο CVS	2,000,000
Μέσος αριθμός <i>commits</i> (σύνολο) ανά ημέρα	Περίπου 500
Εργαλεία υποστήριξης της ανάπτυξης	CVS, GNATS, mailing list και news site

Η C είναι η γλώσσα που κυριαρχεί στο FreeBSD και με μεγαλύτερη απόσταση από την C++ από ότι είδαμε σε άλλες περιπτώσεις σε αυτό το κεφάλαιο. Είναι ενδιαφέρον να σημειώσουμε ότι ο αριθμός γραμμών κώδικα σε γλώσσα assembly που περιέχονται στο FreeBSD, αντιστοιχεί, από άποψη κλίμακος, με εκείνες που βρίσκουμε στο Linux, αν και ο αριθμός γραμμών κώδικα που περιέχονται στον πυρήνα είναι, συνολικά, μόνο 25.000. Για να ανακεφαλαιώσουμε, θα μπορούσαμε να πούμε ότι στο FreeBSD, υπερτερούν οι πιο κλασσικές γλώσσες που μπορούμε να βρούμε μέσα στο Ελεύθερο Λογισμικό, η C, η Shell και η Perl, ενώ οι άλλες γλώσσες που εξετάσαμε στα πλαίσια άλλων εφαρμογών και projects, σαν την C++, την Java, και την Python, δεν ενσωματώθηκαν.

Πίνακας 7. Γλώσσες προγραμματισμού σε χρήση στο FreeBSD

Γλώσσα Προγραμματισμού	Γραμμές Κώδικα	Ποσοστό
C	7.080.000	92.0%
Shell	205.000	2.7%
C++	131.500	1.7%
Assembler	116,000	1.5%
Perl	90,900	1.20%
Yacc	5,800	0.75%

9.2.7. Ακαδημαϊκές μελέτες στο FreeBSD

Αν και είναι αναμφισβήτητα ένα πολύ ενδιαφέρον project (μπορούμε να δούμε την ιστορία του αναλύοντας το σύστημα ελέγχου εκδόσεων, ανατρέχοντας μέχρι και 10 χρόνια πίσω!), το FreeBSD δεν ενέπνευσε τόσο μεγάλο ενδιαφέρον στην επιστημονική κοινότητα. Υπάρχει, πάντως, μία ομάδα έρευνας που επέδειξε ενδιαφέρον για το FreeBSD project, από διάφορες απόψεις ("Incremental and decentralised integration in FreeBSD") [149], και που εστιάσθηκε ιδιαίτερα στο πως τα προβλήματα ενσωμάτωσης του software μπορούν να επιλυθούν με ένα τρόπο επαυξητικό (incremental) και αποκεντρωμένο.

9.3. Το KDE

Αν και δεν επρόκειτο για τη πρώτη λύση, από τη σκοπιά των φιλικών προς τον χρήστη περιβαλλόντων εργασίας, η διάδοση του λειτουργικού συστήματος Windows 95 περί τα μέσα του 1995, οδήγησε σε μία ριζική αλλαγή του τρόπου με τον οποίο οι μη-εξειδικευμένοι χρήστες αλληλεπιδρούν με τους υπολογιστές. Από τα μονοδιάστατα συστήματα με τις γραμμές εντολών (τα τερματικά), γεννήθηκε η μετάβαση στα δύο-διαστάσεων περιβάλλοντα επιφάνειας εργασίας, όπου το ποντίκι άρχισε να χρησιμοποιείται περισσότερο από το πληκτρολόγιο. Πρέπει να αναγνωρίσουμε στα Windows 95, όχι τόσο τις τεχνολογικές καινοτομίες, αλλά το γεγονός ότι υπήρξαν το σύστημα που κατάφερε να κυριαρχήσει τόσο στα προσωπικά όσο και στα εταιρικά περιβάλλοντα εργασίας, καθορίζοντας τις προδιαγραφές που έμελλε να ακολουθηθούν από εκείνο το σημείο και πέρα (τεχνικούς και κοινωνικούς κανόνες από τους οποίους, όπως θα δούμε, υποφέρουμε, σε ορισμένες περιπτώσεις, ακόμη και σήμερα, στις αρχές του 21ου Αιώνα).

Πριν δημιουργηθούν τα συστήματα με περιβάλλοντα επιφάνειας εργασίας, η κάθε εφαρμογή καθόριζε αυτόνομα τον τρόπο της εμφάνισης και της αλληλεπίδρασής της με τον χρήστη. Στα περιβάλλοντα επιφάνειας εργασίας, πάντως, οι εφαρμογές πρέπει να έχουν κοινές ιδιότητες και μία εμφάνιση που να τη μοιράζεται με τις άλλες εφαρμογές, πράγμα που θα μειώνει την πίεση πάνω στον χρήστη, ο οποίος θα μπορεί να ξαναχρησιμοποιεί τον τρόπο αλληλεπίδρασης που έμαθε με μία εφαρμογή, ακόμη και χρησιμοποιώντας άλλες, διαφορετικές εφαρμογές. Αυτό μείωσε την πίεση και πάνω στους developers των εφαρμογών, καθώς δεν έπρεπε πλέον να βρίσκονται κάθε φορά αντιμέτωποι με την αναδημιουργία εκ του μηδενός όλων των διαδραστικών στοιχείων (που αποτελεί, πάντοτε, ένα σύνθετο έργο), αλλά μπορούσαν πλέον να αρχίζουν από ένα προκαθορισμένο framework και προκαθορισμένους κανόνες.

9.3.1. Η ιστορία του KDE

Οι υποστηρικτές του Unix διαπίστωσαν γρήγορα την εξαιρετική επιτυχία των Windows 95 και με δεδομένο ότι τα Unix-οειδή περιβάλλοντα δεν διέθεταν συστήματα που να είναι τόσο απλά και ενστικτώδη στη χρήση (intuitive), ενώ παρέμεναν πάντα ελεύθερα, αποφάσισαν να στρωθούν στη δουλειά. Το KDE K Desktop Environment project γεννήθηκε από αυτή την προσπάθεια το 1996. Σχεδιάστηκε από τον Matthias Ettrich (δημιουργός του LyX, ένα πρόγραμμα επεξεργασίας του TeX typeset) και από άλλους hackers. Το KDE Project πρότεινε τους εξής στόχους:

- Να παρέχει στα Unix-οειδή συστήματα ένα περιβάλλον εργασίας φιλικό προς τον χρήστη που να είναι, συγχρόνως, ανοικτό, σταθερό, αξιόπιστο και ισχυρό.
- Να αναπτύξει ένα σύνολο βιβλιοθηκών για να μπορούν να γραφτούν standard εφαρμογές σε ένα σύστημα γραφικών για Unix X11.
- Να δημιουργήσει μία σειρά εφαρμογών που να επιτρέπουν στον χρήστη να πετύχει τους σκοπούς του με αποτελεσματικό τρόπο.

Όταν τα μέλη του, τότε νεοσύστατου, KDE project αποφάσισαν να χρησιμοποιήσουν μία αντικειμενοστραφή βιβλιοθήκη, ονόματι Qt, προέκυψε μία διαμάχη, καθώς η βιβλιοθήκη ανήκε στην Νορβηγική εταιρεία Trolltech, και δεν καλυπτόταν από κανενός είδους Ελεύθερης Άδειας. Κατά συνέπεια, παρ' ότι οι εφαρμογές KDE υπαγόταν στην άδεια GPL, επειδή συνδεόταν με αυτή την βιβλιοθήκη, σήμαινε ότι ήταν αδύνατον να αναδιανεμηθούν. Έτσι, παραβιαζόταν η μία από τις τέσσερις ελευθερίες που καθιέρωσε ο Richard Stallman στο περίφημο Free Software Manifesto [117]. Από την έκδοση 2.0, η Trolltech διένειμε την Qt υπό μία διπλή άδεια που διευκρίνιζε ότι αν η εφαρμογή, που χρησιμοποιεί την βιβλιοθήκη, λειτουργεί υπό την GPL, τότε η άδεια που θα ισχύει για την Qt θα είναι η GPL. Χάρη σε αυτό, είχαμε ένα ευτυχές τέλος σε μία από τις πιο καυτές και πολυσυζητημένες διαμάχες στον κόσμο του Ελεύθερου Λογισμικού.

Σημείωση

Αρχικά, το όνομα KDE σήμαινε Kool Desktop Environment, αλλά στη συνέχεια άλλαξε απλώς σε KDesktop Environment. Η επίσημη εξήγηση ήταν ότι αυτό συνέβη διότι στο Λατινικό αλφάβητο, το γράμμα K έρχεται ακριβώς πριν το L, στη λέξη Linux.

9.3.2. Ανάπτυξη του KDE

Το KDE είναι ένα από τα λίγα projects Ελεύθερου Λογισμικού που ακολουθεί, σε γενικές γραμμές, ένα χρονοδιάγραμμα για το λανσάρισμα νέων εκδόσεων (ας θυμηθούμε, για παράδειγμα, ότι μία νέα έκδοση του πυρήνα Linux βγαίνει “όταν είναι έτοιμη”, ενώ, όπως θα δούμε και στη συνέχεια, το GNOME πάντα υπέφερε από σημαντικές καθυστερήσεις όταν επρόκειτο να κυκλοφορήσουν νέες εκδόσεις). Η αρίθμηση των νέων εκδόσεων ακολουθεί μία απολύτως καθορισμένη τακτική. Οι εκδόσεις του KDE έχουν 3 αριθμούς εκδόσεως: έναν μεγαλύτερο και 2 μικρότερους. Για παράδειγμα, στο KDE 3.1.2, ο μεγαλύτερος αριθμός είναι το 3, ενώ το 1 και το 2 είναι οι μικρότεροι αριθμοί. Οι εκδόσεις με ταυτόσημους μεγάλους αριθμούς, διαθέτουν δυαδική συμβατότητα, που σημαίνει ότι δεν είναι απαραίτητη η μεταγλώττιση των εφαρμογών. Μέχρι τώρα, οι αλλαγές των μεγαλύτερων αριθμών γινόταν συγχρόνως με τις αλλαγές της βιβλιοθήκης Qt, πράγμα που δείχνει πως οι developers θέλησαν να εκμεταλλευθούν τις νέες λειτουργίες της βιβλιοθήκης Qt, στην επερχόμενη έκδοση του KDE.

Σε ότι αφορά τους μικρότερους αριθμούς, οι εκδόσεις με έναν μόνο μικρότερο αριθμό είναι εκείνες που όχι μόνον περιλαμβάνουν όλες τις νέες λειτουργίες, αλλά ενσωματώνουν και όλες τις διορθώσεις των bugs που είχαν ανακαλυφθεί. Οι εκδόσεις που έχουν και ένα δεύτερο μικρότερο αριθμό, δεν περιλαμβάνουν νέες λειτουργίες συγκριτικά με τις εκδόσεις με τον πρώτο κατώτερο αριθμό, αλλά περιέχουν μόνο τις διορθώσεις των bugs. Το ακόλουθο παράδειγμα θα ξεκαθαρίσει αυτές τις διαφορές καλύτερα: το KDE 3.1 είναι μία έκδοση 3^{ης} γενιάς του KDE (το 3 είναι ο υψηλός αριθμός), στην οποία προστέθηκαν νέες λειτουργίες, ενώ το KDE 3.1.1 είναι η προηγούμενη έκδοση με τις ίδιες λειτουργίες, αλλά με όλα τα bugs που έχουν βρεθεί διορθωμένα.

Αμέσως μετά την έναρξη του project, ένας Σύλλογος με έδρα στη Γερμανία (KDE e.V.) έφτιαξε το KDE και, ως εκ του καταστατικού του, προβλεπόταν ότι πρέπει να υπάρχει μία Διοικούσα Επιτροπή. Η επιρροή αυτού του Διοικητικού οργάνου πάνω στην ανάπτυξη είναι αμελητέα, καθώς η βασική του ασχολία είναι η διοίκηση του Συλλόγου, ειδικά σε ότι αφορά τις δωρεές που γίνονται προς το project. Για να προωθηθεί και να διαδοθεί το KDE, δημιουργήθηκε η KDE League, που περιλαμβάνει όλες τις εμπλεκόμενες εταιρείες, όπως θα δούμε παρακάτω.

9.3.3. Η KDE League

Η KDE League είναι μία ομάδα εταιρειών και φυσικών προσώπων από το KDE, που έχουν ως στόχο την προώθηση, την διανομή και την ανάπτυξη του KDE. Οι εταιρείες και τα άτομα που συμμετέχουν στην KDE League δεν χρειάζεται να εμπλέκονται άμεσα στην ανάπτυξη του KDE (αν και τα μέλη ενθαρρύνονται να συμμετάσχουν), παρά απλώς αντιπροσωπεύουν ένα βιομηχανικό και κοινωνικό πλαίσιο (framework) που είναι φιλικό προς το KDE. Οι σκοποί της KDE League είναι οι εξής:

- Η προώθηση, η παροχή και η διευκόλυνση επίσημης και ανεπίσημης εκπαίδευσης γύρω από τις λειτουργίες, τις ικανότητες και τις άλλες ποιότητες του KDE.
- Η ενθάρρυνση εταιρειών, κυβερνήσεων, επιχειρήσεων και ατόμων να χρησιμοποιούν το KDE.
- Η ενθάρρυνση όλων των παραπάνω να συμμετέχουν στην ανάπτυξη του KDE.
- Η παροχή γνώσης, πληροφορίας, διαχείρισης και υιοθέτησης του KDE, από τη σκοπιά της χρήσης και της ανάπτυξής του.
- Η καλλιέργεια κλίματος καλής επικοινωνίας και συνεργασίας στους developers του KDE.
- Η καλλιέργεια κλίματος καλής επικοινωνίας και συνεργασίας ανάμεσα στους KDE developers και το ευρύ κοινό, με τη βοήθεια δημοσιευμάτων, άρθρων, ιστοτόπων, συναντήσεων, συμμετοχής σε συνέδρια και εκθέσεις, άρθρων στα MME, συνεντεύξεων, προωθητικού υλικού και επιτροπών.

Οι εταιρείες που συμμετέχουν στην KDE League είναι κυρίως σχεδιαστές διανομών (η SuSE, τώρα μέρος της Novell, η Mandriva, το TurboLinux, τα Linux και η Hancorn, μία Κορεατική διανομή Ελεύθερου Λογισμικού), εταιρείες ανάπτυξης (η Trolltech και η Klarälvdalens Datakonsult AB), και επιπλέον ο γίγαντας IBM καθώς και μία εταιρεία που δημιουργήθηκε με στόχο την προώθηση του KDE (KDE.com). Από όλες αυτές, πρέπει να αναφερθούμε ιδιαιτέρως στην Trolltech, την Novell και την Mandriva Software, των οποίων η εμπλοκή υπήρξε κρίσιμης σημασίας και των οποίων τα επιχειρηματικά μοντέλα συνδέονται πολύ στενά με το KDE project:

- Η Trolltech είναι μία Νορβηγική εταιρεία με έδρα στο Όσλο, που αναπτύσσει την Qt, μία βιβλιοθήκη που μπορεί να χρησιμοποιηθεί ως γραφική διεπαφή για τον χρήστη και ως ένα API για developers, αν και μπορεί να λειτουργήσει και ως ένα στοιχείο ενσωματωμένο σε μία συσκευή PDA (σαν το Sharp Zaurus). Η σημασία του KDE project για την Trolltech είναι φανερή από δύο βασικά στοιχεία της εμπορικής της τακτικής: από τη μία μεριά, αναγνωρίζει το KDE ως την κύρια μέθοδο προώθησής της, ενθαρρύνοντας την ανάπτυξη του περιβάλλοντος για την επιφάνεια εργασίας, κάνοντας αποδεκτές και υλοποιώντας τις προταθείσες βελτιώσεις ή τροποποιήσεις, ενώ από την άλλη μεριά, μερικοί από τους πιο σημαντικούς KDE developers εργάζονται επαγγελματικά για την Trolltech. Το πιο γνωστό παράδειγμα είναι εκείνο του ίδιου του Matthias Ettrich, που υπήρξε και ο ιδρυτής του project, και ο οποίος είναι αναμφισβήτητα χρήσιμος τόσο για το KDE project όσο και για την ίδια την εταιρεία. Η ανάμειξη της Trolltech στο KDE project δεν περιορίζεται αυστηρά στη βιβλιοθήκη Qt, όπως αποδεικνύεται από το γεγονός ότι ένας από τους βασικούς developers του Koffice (ένα πακέτο λογισμικού του KDE που προσφέρει μία σουίτα γραφείου), έχει, τρεχόντως, μία σύμβαση part-time μαζί τους.

- Το SuSE (τώρα μέρος της Novell) έδειχνε πάντα την ιδιαίτερη προτίμησή του για το σύστημα με το περιβάλλον KDE στην επιφάνεια εργασίας, εν μέρει οφειλόμενη στο γεγονός πως οι περισσότεροι από τους δικούς του developers είναι είτε Γερμανικής είτε Κεντρο- Ευρωπαϊκής καταγωγής, όπως και η ίδια η εταιρεία. Η SuSE, συνειδητοποιώντας το γεγονός ότι όσο καλύτερο και ευκολότερο είναι το γραφικό περιβάλλον εργασίας που προσφέρει η διανομή της, τόσο μεγαλύτερη θα είναι και η αποδοχή του και αντίστοιχα υψηλές θα είναι, κατά συνέπεια, και οι πωλήσεις και τα αιτήματα για υποστήριξη, εφάρμοζε πάντα μία πολύ ενεργή στρατηγική σε ότι αφορά τα κονδύλια που αφιέρωνε στην υποστήριξη του επαγγελματισμού των βασικών θέσεων μέσα στο KDE project. Για παράδειγμα, ο τωρινός διαχειριστής του συστήματος ελέγχου των εκδόσεων και ακόμη δύο από τους βασικούς developers, είναι όλοι στο μισθολόγιο της SuSE. Αντιστοίχως, μέσα στο ανθρώπινο δυναμικό της, υπάρχουν καμία δωδεκαριά developers που μπορούν να ξοδεύουν μέρος του εργάσιμου χρόνου τους για την ανάπτυξη του KDE.

- Η διανομή Mandriva είναι ακόμη ένας από τους μεγαλύτερους υποστηρικτές του KDE και αρκετοί από τους βασικούς developers εργάζονται γι' αυτό. Η οικονομική της κατάσταση, όμως, που έφθασε ως την πτώχευση το 2003, σήμανε ότι έχασε την επιρροή της κατά τα επόμενα λίγα χρόνια.

9.3.4. Η παρούσα κατάσταση του KDE

Μετά την δημοσίευση του KDE 3, τον Μάιο του 2002, η γενική άποψη που σχηματίστηκε ήταν ότι τα ελεύθερα γραφικά περιβάλλοντα εργασίας είναι ισότιμα με τους ιδιοταγείς ανταγωνιστές τους. Μερικά από τα μεγαλύτερα επιτεύγματά του περιλαμβάνουν την ενσωμάτωση ενός συστήματος επί μέρους συστατικών (components system), του Kparts, που καθιστά εφικτή την ενσωμάτωση κάποιων εφαρμογών μέσα σε άλλες (ένα κομμάτι του λογιστικού φύλλου Kspread, μέσα στον επεξεργαστή κειμένου KWord) και την ανάπτυξη του DCOP, ενός απλού συστήματος για την εύκολη επικοινωνία των διαφόρων διαδικασιών (processes) μεταξύ τους, με authentication. Το DCOP αποτέλεσε τη δέσμευση του project που έδρασε βλαπτικά προς τις τεχνολογίες CORBA, ένα πολυσυζητημένο θέμα στον κόσμο των ελεύθερων γραφικών διεπαφών για την επιφάνεια εργασίας, ειδικά για το GNOME, όπου αποφασίσθηκε να χρησιμοποιηθεί το CORBA.

Η ιστορία μοιάζει να έχει βάλει στη θέση της την κάθε τεχνολογία, όπως μπορούμε να δούμε από την πρόταση χρήσης του DBUS (ένα είδος βελτιωμένου DCOP) εκ μέρους του FreeDesktop.org,

ενός project που ενδιαφερόταν για την προώθηση της διαλειτουργικότητας και τη χρήση των σχετικών τεχνολογιών στις ελεύθερες γραφικές διεπαφές για την επιφάνεια εργασίας και του οποίου ηγείται, κατά σύμπτωση, ένας από τους πιο γνωστούς GNOME *hackers*.

Ο παρακάτω πίνακας ανακεφαλαιώνει τα πιο σημαντικά χαρακτηριστικά του KDE project. Οι Άδειες που γίνονται δεκτές από το project εξαρτώνται από το εάν είναι για μία εφαρμογή ή μία βιβλιοθήκη. Οι άδειες των βιβλιοθηκών παρέχουν μεγαλύτερη "ευελιξία" προς τους τρίτους. Με άλλα λόγια, καθιστούν δυνατόν σε τρίτους ενδιαφερόμενος, να δημιουργούν ιδιοταγείς εφαρμογές που να συνδέονται με τις βιβλιοθήκες.

Η πιο πρόσφατη έκδοση του KDE είναι, στις αρχές του 2007, η 3.5.6 και αποτελεί την τέταρτη γενιά, ενώ το KDE 4, που θα βασίζεται στην Qt4, αναμένεται να φθάσει κατά τα μέσα του 2007. Η αλλαγή γενιάς περιλαμβάνει πολλή προσπάθεια προσαρμογής της τρέχουσας έκδοσης, πράγμα που είναι μία πολύ πληκτική και χρονοβόρα διαδικασία. Αυτό, βέβαια, δεν σημαίνει ότι οι "παλιές" εφαρμογές δεν θα δουλεύουν πλέον. Γενικά, για να διατηρηθούν σε λειτουργία, περιλαμβάνονται και οι παλιότερες εκδόσεις των βιβλιοθηκών, πάνω στις οποίες βασίζονταν, παρά το ότι αυτό σημαίνει πως πρέπει να φορτώνονται συγχρόνως διάφορες εκδόσεις βιβλιοθηκών στη μνήμη, με την επακόλουθη σπατάλη των πόρων του συστήματος. Οι developers του KDE το θεωρούν ως ένα ενδογενές μέρος της διαδικασίας ανάπτυξης του project και, συνεπώς, σαν ένα έλασσον κακό.

9.3.5. Ακτινογραφία του KDE

Σε ότι αφορά την κλίμακα του KDE, θα συζητήσουμε τώρα κάποιους αριθμούς που αντιστοιχούν στην κατάσταση του CVS, κατά τον Αύγουστο του 2003, που σημαίνει ότι θα πρέπει να ερμηνεύονται με τις ίδιες συνήθειες επιφυλάξεις που ήδη αναλύσαμε, συν μία επιπλέον: μερικές από τις ενότητες/modules που χρησιμοποιήθηκαν σε αυτή τη μελέτη, βρίσκονται ακόμη υπό ανάπτυξη και δεν πληρούν τα κριτήρια για να λογίζονται ως τελειωμένο προϊόν. Αυτό δεν θα πρέπει να έχει καμία επίδραση σε ότι αφορά τους σκοπούς μας, καθώς ενδιαφέρει περισσότερο η κλίμακα των αποτελεσμάτων, παρά οι ακριβείς αριθμοί.

Ο πηγαίος κώδικας που περιλαμβάνεται στο CVS του KDE, είναι το συνολικό άθροισμα έξι εκατομμυρίων γραμμών κώδικα, σε διάφορες γλώσσες προγραμματισμού, όπως θα δείξουμε πιο κάτω. Ο υπολογιζόμενος χρόνος που απαιτείται για τη δημιουργία του KDE θα έπρεπε να ήταν περίπου εννιάμισι (9.5) χρόνια, που είναι παραπάνω από τα επτά (7) χρόνια που χρειάστηκε το project, και ο μέσος εκτιμώμενος αριθμός των developers που θα πρέπει να εργάζονται full-time, θα έπρεπε να είναι διακόσιοι (200). Αν λάβουμε υπ' όψη το γεγονός πως το KDE διέθετε περίπου 800 ανθρώπους με δικαίωμα write-access στο CVS το 2003 (εκ των οποίων οι μισοί ήταν αδρανείς για περισσότερο από τα δύο τελευταία χρόνια) και το δεδομένο ότι ο αριθμός των KDE developers με συμβάσεις αποκλειστικής απασχόλησης (*full-time*) δεν ήταν παραπάνω από 20, σε οποιαδήποτε χρονική στιγμή, τότε μπορούμε να πούμε ότι το επίπεδο παραγωγικότητας του KDE είναι πολύ, πολύ μεγαλύτερο από εκείνο που προβλέπει θεωρητικά το μοντέλο COCOMO.

Σημείωση

Μία εταιρεία που θέλησε να αναπτύξει ένα προϊόν αυτής της κλίμακας, αρχίζοντας από το μηδέν, θα έπρεπε να επενδύσει παραπάνω από 250 εκατομμύρια δολάρια. Προς χάριν συγκρίσεως, αυτό το ποσό θα ήταν ισοδύναμο με την επένδυση μίας αυτοκινητοβιομηχανίας για το στήσιμο ενός νέου εργοστασίου παραγωγής στην Ανατολική Ευρώπη ή με την πρόβλεψη δαπανών μίας πετρελαϊκής βιομηχανίας για να ανοίξει 200 βενζινάδικα στην Ισπανία.

Είναι, επίσης, ενδιαφέρουσα η διαπίστωση ότι μεγάλο μέρος της προσπάθειας αυτής, σχεδόν το

ήμισυ εκείνης που επενδύθηκε για την ανάπτυξη του DE project, αντιστοιχεί στη μετάφραση της διεπαφής του χρήστη και της συνοδευτικής τεκμηρίωσης (εγχειρίδια, κλπ.). Αν και πολύ λίγες (περίπου 1000) από τις γραμμές προγραμματισμού ασχολούνται με αυτή τη δουλειά, ο αριθμός αρχείων που είναι αφιερωμένα σε αυτό το σκοπό είναι 75.000 μεταφράσεις (αριθμός που αυξάνεται στις 100.000, αν συμπεριλάβουμε τη γραπτή τεκμηρίωση στις διάφορες μορφές της). Αυτό περιλαμβάνει σχεδόν το ένα τέταρτο (ή ένα τρίτο) από τα 310.000 αρχεία που υπάρχουν στο CVS. Η συνδυασμένη δραστηριότητα του CVS ανέρχεται στα 1200 *commits* ανά ημέρα, που σημαίνει ότι ο μέσος χρόνος ανάμεσα σε δύο *commits* είναι περίπου της τάξης του ενός λεπτού ⁽¹⁰⁾.

Σε ότι αφορά τα εργαλεία, τις τοποθεσίες για την πληροφορία και την υποστήριξη για την ανάπτυξη, θα δούμε ότι το φάσμα δυνατοτήτων που προσφέρει το KDE είναι κατά πολύ ευρύτερο από εκείνο που χρησιμοποιείται στο Linux. Εκτός από το σύστημα ελέγχου εκδόσεων και τις mailing lists, το KDE διαθέτει μια σειρά ιστοσελίδων που παρέχουν πληροφόρηση και τεχνική καθώς και μη-τεχνική τεκμηρίωση σχετικά με το project. Ανάμεσα σε αυτούς τους ιστότοπους, υπάρχει μία νέα ιστοσελίδα όπου παρουσιάζονται οι νέες λύσεις και όπου συζητούνται οι διάφορες προτάσεις. Αυτή η ιστοσελίδα με όλα τα νέα, δεν μπορεί να θεωρηθεί σε καμία περίπτωση σαν ένα υποκατάστατο για τις mailing lists, οι οποίες, όπως και στην περίπτωση του Linux, είναι το μέρος όπου λαμβάνουν χώρα οι πραγματικές συζητήσεις και όπου λαμβάνονται οι αποφάσεις και καθορίζονται οι μελλοντικές στρατηγικές. Η ιστοσελίδα με τα νέα είναι, βασικά, το σημείο όπου συναντιούνται οι χρήστες. Τέλος, το KDE οργανώνει τακτικές συναντήσεις, εδώ και 3 χρόνια, όπου συναντιούνται οι developers και οι συνεργάτες, για περίπου 1 εβδομάδα, για να παρουσιάσουν τις τελευταίες καινοτομίες, να συζητήσουν, να γνωριστούν μεταξύ τους και να περάσουν όμορφα (όχι απαραίτητα με αυτή τη σειρά).

⁽¹⁰⁾ Θα πρέπει, σε αυτό το σημείο, να κάνουμε δύο παρατηρήσεις: η πρώτη είναι ότι, όταν γίνεται ένα *commit* που περιλαμβάνει διάφορα αρχεία, είναι σαν να είχε γίνει από ένα ξεχωριστό *commit* για το κάθε αρχείο. Η δεύτερη παρατήρηση είναι ότι ο αριθμός των *commits* είναι ένα κατ' εκτίμηση υπολογισμένο ποσό, καθώς το project διαθέτει μία σειρά από scripts που εκτελούν αυτόματα τα *commits*.

Πίνακας 8. Ανάλυση του KDE

Ιστότοπος	http://www.kde.org
Έναρξη του project	1996
Άδεια (για τις εφαρμογές)	GPL, QPL, MIT, Artistic
Άδεια (για τις βιβλιοθήκες)	LGPL, BSD, X11
Έκδοση που αναλύθηκε	3.1.3
Γραμμές πηγαίου κώδικα	6.100.000
Αριθμός αρχείων (κώδικας, τεκμηρίωση, κλπ.)	310.000 αρχεία
Εκτίμηση κόστους	\$ 255.000.000
Εκτίμηση απαιτούμενου χρόνου ως την ολοκλήρωση	9.41 έτη (112.98 μήνες)
Εκτίμηση μέσου αριθμού developers	200.64
Κατά προσέγγιση αριθμός developers	Περίπου 900 <i>committers</i>
Αριθμός δραστήριων <i>committers</i> κατά το τελευταίο έτος	Γύρω στους 450 (περίπου το 50% του συνόλου)
Αριθμός δραστήριων <i>committers</i> κατά τα	Γύρω στους 600 (περίπου το 65% του συνόλου)

τελευταία 2 χρόνια	
Κατά προσέγγιση αριθμός μεταφραστών (δραστήριοι)	Περίπου 300 μεταφραστές για πάνω από 50 γλώσσες (περιλαμβανομένης της Esperanto).
Αριθμός <i>commits</i> στο CVS (από τους developers)	Περίπου 2.000.000 (εκτιμώμενος αριθμός, χωρίς τα αυτόματα <i>commits</i>)
Αριθμός <i>commits</i> στο CVS (από τους μεταφραστές)	Περίπου 1.000.000 (εκτιμώμενος αριθμός, χωρίς τα αυτόματα <i>commits</i>)
Μέσος αριθμός <i>commits</i> (σύνολο) ανά ημέρα	1.700
Εργαλεία, τεκμηρίωση και εκδηλώσεις υποστήριξης της ανάπτυξης	CVS, mailing lists, ιστοσελίδα, news site, Ετήσιες συναντήσεις

Σε ότι αφορά τις γλώσσες προγραμματισμού που χρησιμοποιούνται στο KDE, η C++ κυριαρχεί. Αυτό οφείλεται κυρίως στο γεγονός ότι αυτή είναι η ενδογενής (native) γλώσσα της Qt, αν και καταβάλλονται μεγάλες προσπάθειες για να προσφερθούν ανοίγματα (*bindings*) που να επιτρέπουν την ανάπτυξη και σε άλλες γλώσσες προγραμματισμού. Σίγουρα, ο αριθμός γραμμών κώδικα στις γλώσσες των διαφόρων μειονοτήτων αντιστοιχεί σχεδόν καθ' ολοκληρίαν σε αυτά τα *bindings*, παρόλο που αυτό δεν σημαίνει ότι αυτές δεν χρησιμοποιούνται καθόλου, μιας και υπάρχουν πολυάριθμα projects που λειτουργούν εξωτερικά σε σχέση με το KDE και που τις χρησιμοποιούν.

Πίνακας 9. Προγραμματιστικές γλώσσες που χρησιμοποιούνται στο KDE

Προγραμματιστική γλώσσα	Γραμμές κώδικα	Ποσοστό
C++	5.011.288	82.05%
C	575.237	9.42%
Objective C	144.415	2.36%
Shell	103.132	1.69%
Java	87.974	1.44%
Perl	85.869	1.41%

9.4. Το GNOME

Ο βασικός σκοπός του GNOME project είναι να δημιουργήσει ένα σύστημα για την επιφάνεια εργασίας για τον τελικό χρήστη, που να είναι πλήρες, ελεύθερο και εύχρηστο. Αντιστοίχως, η ιδέα είναι να μπορεί το GNOME να αποτελέσει μία πολύ ισχυρή πλατφόρμα για τους developers. Τα αρχικά GNOME σημαίνουν *GNU Network Object Model Environment*. Από το όνομά του, βλέπουμε ότι το GNOME αποτελεί μέρος του GNU project. Προς το παρόν, όλος ο κώδικας που υπάρχει μέσα στο GNOME πρέπει να είναι υπό την Άδεια GNU GPL ή την GNU LGPL. Μπορούμε, επίσης, να αντιληφθούμε ότι τα δίκτυα (*networks*) και η αντικειμενοστρεφής μοντελοποίηση (*object-orientated modelling*) είναι εξαιρετικά σημαντικά στοιχεία.

9.4.1. Η ιστορία του GNOME

Ενώ συνεχιζόταν ακόμη η διαμάχη περί της της ελευθερίας του KDE, το καλοκαίρι του 1997, μία συγκυρία οδήγησε στη συνάντηση του Miguel de Icaza με τον Nat Friedman, στο Redmond, κατά τη διάρκεια κάποιων workshops που είχε οργανώσει η Microsoft. Είναι πιθανόν αυτή η συνάντηση να προκάλεσε μία ριζική μεταστροφή και στους δύο, οδηγώντας στην δημιουργία του GNOME από

τον Miguel de Icaza όταν επέστρεψε στο Μεξικό (μαζί με τον Federico Mena Quintero) και το θαυμασμό του για τις κατανομημένες αντικειμενοστρεφείς τεχνολογίες (*distributed object technologies*). Ο De Icaza και ο Mena αποφάσισαν να φτιάξουν ένα περιβάλλον που να είναι μία εναλλακτική για το KDE, καθώς κατάλαβαν ότι η εκ νέου υλοποίηση μίας ιδιοταγούς βιβλιοθήκης θα ήταν ένα έργο με προδιαγεγραμμένη αποτυχία. Έτσι, λοιπόν, γεννήθηκε το GNOME.

Από εκείνους τους παλιούς καιρούς, το 1997, το GNOME μεγάλωσε σταδιακά και συνεχίζει να αναπτύσσεται, με τις αλλεπάλληλες εκδόσεις του. Η έκδοση 0.99 κυκλοφόρησε το Νοέμβριο του 1998, αλλά η πρώτη πραγματικά δημοφιλής έκδοση, που διενέμετο ουσιαστικά με κάθε διανομή GNU/Linux, έμελλε να είναι το GNOME 1.0, που δημοσιοποιήθηκε το Μάρτιο του 1999. Θα πρέπει να σημειωθεί ότι η εμπειρία από αυτή την πρώτη σταθερή έκδοση του GNOME δεν ήταν και τόσο ικανοποιητική, καθώς για πολλούς ήταν γεμάτη με κρίσιμης σημασίας bugs. Για τον λόγο αυτό, το GNOME October (GNOME 1.0.55) θεωρείται ως η πρώτη έκδοση του περιβάλλοντος εργασίας GNOME που ήταν πραγματικά σταθερή. Όπως μπορούμε να παρατηρήσουμε, με το GNOME October, οι developers δεν χρησιμοποιούν αριθμημένες εκδόσεις, για να αποφύγουν να μπουν σε μία “κούρσα ανταγωνισμού” με τις εκδόσεις του KDE. Το πρώτο GUADEC, το Ευρωπαϊκό Συνέδριο των χρηστών και των developers του GNOME, έλαβε χώρα στο Παρίσι το 2000 και παρά λίγο να συμπέσει με τη δημοσιοποίηση της νέας έκδοσης του GNOME, που ονομαζόταν GNOME April.

Αυτή ήταν και η τελευταία έκδοση που έπαιρνε το όνομα ενός μηνός, καθώς προέκυψε ότι αυτό το σύστημα οδήγησε σε παραπάνω σύγχυση από οτιδήποτε άλλο (για παράδειγμα, το GNOME April κυκλοφόρησε μετά το GNOME October, οπότε ήταν δικαιολογημένο αν κάποιος έβγαζε το αντίθετο συμπέρασμα). Τον Οκτώβριο του ίδιου έτους, μετά από πολλές αντιπαραθέσεις σχετικά με αυτό το ζήτημα στις διάφορες mailing lists, δημιουργήθηκε το Ίδρυμα GNOME Foundation, το οποίο θα εξετάσουμε στα επόμενα κεφάλαια.

Το GNOME 1.2 απετέλεσε ένα βήμα εμπρός από την άποψη της αρχιτεκτονικής που χρησιμοποιούσε το GNOME, μία δομή που συνέχισε να χρησιμοποιεί και με το GNOME 1.4. Αυτή η εποχή σημαδεύτηκε από το δεύτερο GUADEC, που έλαβε χώρα στην Κοπεγχάγη. Εκείνο που άρχισε σαν μία μικρή συνάντηση για λίγους *hackers*, μετεβλήθη σε ένα μέγα συμβάν που αιχμαλώτισε την προσοχή ολόκληρης της βιομηχανίας λογισμικού.

Στο εν τω μεταξύ, η αντιπαραθεση σχετικά με την ελευθερία του KDE επιλύθηκε μετά την αλλαγή στάσης εκ μέρους της Trolltech, όταν αποφάσισε να θέσει τέρμα στη διπλή αδειοδότηση της Qt.

Σήμερα, δεν υπάρχει αμφιβολία ότι τόσο το GNOME όσο και το KDE είναι αμφότερα ελεύθερα γραφικά περιβάλλοντα για την επιφάνεια εργασίας, πράγμα που μας επιτρέπει να πούμε ότι η ανάπτυξη του GNOME έδωσε ενθάρρυνση για τη δημιουργία όχι μόνο μιας, αλλά δύο ελεύθερων γραφικών διεπαφών για την επιφάνεια εργασίας.

9.4.2. Το Ίδρυμα GNOME Foundation

Όταν ακούς για πρώτη φορά για το GNOME, το πιο δύσκολο να αντιληφθείς είναι το πρόβλημα της οργάνωσης περισσότερων από 1000 συντελεστών (*contributors*) στα πλαίσια του project. Είναι παράδοξο, όταν ένα project με μία δομή που τείνει προς το αναρχικό, σημειώνει όλη αυτή την επιτυχία και κατακτά σύνθετους στόχους, που μόνο λίγες πολυεθνικές στον χώρο του IT θα μπορούσαν να καταφέρουν.

Παρότι το GNOME φτιάχθηκε με σαφή σκοπό να προσφέρει ένα ισχυρό και φιλικό προς τον χρήστη περιβάλλον, στο οποίο, βαθμιαία, θα προστίθενται νέα προγράμματα, έγινε γρήγορα κατανοητό ότι ήταν απαραίτητη η δημιουργία ενός σώματος με συγκεκριμένες ευθύνες που θα του επιτρέπουν να προωθεί και να ενισχύει τη χρήση, την ανάπτυξη και τη διάδοση του GNOME: έτσι, το 2000, δημιουργήθηκε το Ίδρυμα GNOME Foundation, με έδρα στην Βοστώνη των ΗΠΑ.

Το Ίδρυμα GNOME Foundation είναι μία Μ.Κ.Ο. και όχι μία εταιρική κοινοπραξία (*industrial consortium*). Έχει τις εξής λειτουργίες:

- Συντονίζει τις δημοσιεύσεις.
- Αποφασίζει για το ποια projects θα αποτελούν μέρος του GNOME.
- Είναι το επίσημο αντιπροσωπευτικό όργανο (για τα ΜΜΕ και για τις βιομηχανίες) του GNOME project.
- Προωθεί οργάνωση συνεδρίων που σχετίζονται με το GNOME (όπως το GUADEC).
- Αντιπροσωπεύει το GNOME σε άλλα συνέδρια.
- Δημιουργεί τεχνικές προδιαγραφές.
- Προωθεί τη χρήση και την ανάπτυξη του GNOME.

Επιπλέον, το Ίδρυμα GNOME Foundation δέχεται χρηματικά εμβάσματα για την προώθηση και ενίσχυση των παραπάνω λειτουργιών, καθώς κάτι τέτοιο ήταν ανέφικτο να γίνει με διαφανή τρόπο πριν την δημιουργία του Ιδρύματος.

Προς το παρόν, το GNOME Foundation έχει έναν υπάλληλο ολοκληρωτικής απασχόλησης, που είναι υπεύθυνος για την διεκπεραίωση όλων των γραφειοκρατικών και οργανωτικών εργασιών που πρέπει να γίνονται σε μία ΜΚΟ, που έχει συναντήσεις και συνέδρια ανά τακτά διαστήματα.

Σε γενικές γραμμές, το GNOME Foundation περιλαμβάνει δύο μεγάλες επιτροπές: μία Διαχειριστική και μία Συμβουλευτική Επιτροπή. Η Διαχειριστική Επιτροπή (το Δ.Σ.) αποτελείται, κατά πλειοψηφία, από 14 μέλη δημοκρατικά εκλεγμένα από τα μέλη του GNOME Foundation. Ακολουθείται ένα "αξιοκρατικό" μοντέλο, που σημαίνει ότι, για να γίνει κανείς μέλος του GNOME Foundation, πρέπει να έχει συνεργασθεί, με τον ένα ή με τον άλλο τρόπο, με το GNOME project. Η συνεισφορά δεν σημαίνει υποχρεωτικά συγγραφή πηγαίου κώδικα. Υπάρχουν και άλλες εργασίες, όπως η μετάφραση, η οργάνωση, η διάδοση, κλπ., που θα μπορούσε κάποιος να επιτελέσει, για να μπορεί να κάνει αίτηση και να γίνει μέλος του GNOME Foundation, ούτως ώστε να έχει δικαίωμα ψήφου. Συνεπώς, είναι τα ίδια τα μέλη του Foundation που μπορούν να βάζουν υποψηφιότητα για το Δ.Σ. και είναι και πάλι τα ίδια τα μέλη που εκλέγουν δημοκρατικά τους αντιπροσώπους τους στο Δ.Σ. από τους διαθέσιμους υποψηφίους. Στην παρούσα φάση, η ψηφοφορία γίνεται μέσω e-mail. Η διάρκεια του αξιώματος ενός μέλους στο Δ.Σ. ορίζεται στο ένα έτος, και μετά από αυτό προβλέπονται εκ νέου εκλογές.

Υπάρχουν μερικοί βασικοί κανονισμοί για τη διασφάλιση της διαφάνειας στο Δ.Σ. Ο πιο αξιοσημείωτος είναι ο περιορισμός του αριθμού των μελών που είναι ταυτοχρόνως συνδεδεμένοι με την ίδια εταιρεία, και ο οποίος δεν μπορεί να υπερβαίνει τους τέσσερις (4) υπαλλήλους. Είναι σημαντικό να τονίσουμε ότι τα μέλη του Δ.Σ. δρουν πάντοτε υπό την ατομική τους ιδιότητα, και ουδέποτε αντιπροσωπεύοντας την εταιρεία. Παρ' όλα ταύτα, μετά από μεγάλη συζήτηση, συμφωνήθηκε να συμπεριληφθεί και αυτή η ρήτρα για την αποφυγή δυσπιστίας.

Η άλλη επιτροπή μέσα στο GNOME Foundation είναι η Συμβουλευτική Επιτροπή, η οποία δεν έχει τη δυνατότητα της λήψης αποφάσεων, αλλά παίζει τον ρόλο του οχήματος επικοινωνίας με το Δ.Σ. Αποτελείται από εμπορικές εταιρείες που δραστηριοποιούνται στη βιομηχανία λογισμικού, καθώς και από μη-εμπορικές οργανώσεις. Στις μέρες μας, στα μέλη του περιλαμβάνονται η Red Hat, η Novell, η Hewlett-Packard, η Mandrake, η SUN Microsystems, η Red Flag Linux, η Wipro, το Debian και το Free Software Foundation. Όλες οι εταιρείες που έχουν πάνω από 10 υπαλλήλους, πρέπει να καταβάλλουν ένα ποσό, για να είναι μέλη της Συμβουλευτικής Επιτροπής.

9.4.3. Οι βιομηχανίες που εργάζονται γύρω από το GNOME

Το GNOME κατάφερε να διεισδύσει σημαντικά στη βιομηχανία, σε βαθμό που διάφορες εταιρείες συμμετείχαν πολύ ενεργά στην ανάπτυξή του. Από όλες αυτές, οι πιο σημαντικές περιπτώσεις είναι εκείνες της Ximian Inc., της Eazel, της RHAD Labs από την Red Hat και, πιο πρόσφατα, η SUN Microsystems. Θα περιγράψουμε, τώρα, τα κίνητρα των εταιρειών για κάθε μία περίπτωση, καθώς και τις σημαντικότερες συνεισφορές τους στο περιβάλλον επιφάνειας εργασίας του GNOME:

- Η Ximian Inc. (αρχικά αποκαλούμενη Helix Inc.) είναι το όνομα της εταιρείας που ιδρύθηκε το 1999 από τον Miguel de Icaza, τον συνιδρυτή του GNOME, και τον Nat Friedman, έναν από τους *hackers* του GNOME. Ο βασικός της σκοπός ήταν να προσελκύσει τους σημαντικότερους GNOME developers κάτω από την ίδια στέγη, για να μεγιστοποιήσει την ανάπτυξή του, οπότε δεν αποτελεί έκπληξη το γεγονός ότι ανάμεσα στους τωρινούς αλλά και στους παλιότερους υπαλλήλους της συγκαταλέγονται περίπου 20 από τους πλέον ενεργούς GNOME developers. Η εφαρμογή στην οποία η Ximian έβαλε την περισσότερη προσπάθεια ήταν, εξ' αρχής, το Evolution, ένα πλήρες σύστημα διαχείρισης προσωπικών πληροφοριών (Personal Information Management), στο στυλ του Microsoft Outlook, που συμπεριελάμβανε έναν e-mail client, μία ατζέντα και ένα βιβλίο διευθύνσεων για τις επαφές μας. Τα προϊόντα που πωλούσε η Ximian ήταν το Ximian Desktop (μία έκδοση του GNOME με εταιρική στόχευση), το Red Carpet (που ήταν κυρίως για το σύστημα διανομής λογισμικού GNOME -αν και δεν περιοριζόταν μόνο σε αυτό- και, τέλος, το MONO (μία νέα υλοποίηση της πλατφόρμας .NET), αν και η τελευταία δεν σχετίζεται, προς το παρόν, κατά κανέναν τρόπο με το GNOME. Η Ximian ανέπτυξε, επίσης, μία εφαρμογή που επιτρέπει στο Evolution να επικοινωνεί με έναν Exchange 2000 server. Αυτή η εφαρμογή, παρότι είναι ακόμη πολύ μικρή, αποδείχθηκε πολύ αμφιλεγόμενη διότι κυκλοφόρησε υπό μία μη-ελεύθερη άδεια (ακολουθώντας, το 2004, και αυτό το προϊόν αδειοδοτήθηκε ως Ελεύθερο Λογισμικό). Τον Αύγουστο του 2003, η Novell, στο πλαίσιο της στρατηγικής για τη διείσδυσή της στο περιβάλλον της επιφάνειας εργασίας του GNU/Linux, εξαγόρασε την Ximian.

- Η Eazel ιδρύθηκε το 1999 από μία ομάδα ανθρώπων που εργαζόταν για την Apple, με στόχο να κάνουν το περιβάλλον του GNU/Linux το ίδιο εύκολο σαν το περιβάλλον της Macintosh. Η εφαρμογή πάνω στην οποία συγκέντρωσαν τις προσπάθειές τους λεγόταν Nautilus (Ναυτίλος) και υποτίθεται ότι θα γινόταν ο διαχειριστής αρχείων που θα έστελνε οριστικά στη σύνταξη τον μυθικό Midnight Commander, που είχε αναπτύξει ο Miguel de Icaza. Η έλλειψη ενός επιχειρηματικού μοντέλου, σε συνδυασμό με την κρίση των εταιρειών .dotcom, οδήγησε τους ριψοκίνδυνους επενδυτές να αποσύρουν όλα τα κεφάλαια που ήταν απαραίτητα για τη συνέχιση των δραστηριοτήτων της εταιρείας, με αποτέλεσμα η Eazel να δηλώσει πτώχευση στις 15/05/2001 και να κατεβάσει τα στόρια της. Πριν από αυτό, είχε, παρ' όλα αυτά, το χρόνο να κυκλοφορήσει τον Ναυτίλο, στην έκδοση 1.0, αν και η αρίθμηση ήταν μάλλον παραπλανητική, αφού η σταθερότητα που κάποιος θα περίμενε από μία τέτοια έκδοση 1.0, δεν φαινόταν πουθενά. Δύο χρόνια μετά την πτώχευση της Eazel, μπορούμε να διαπιστώσουμε πως ο Ναυτίλος είχε αναπτυχθεί και είχε γίνει ένας πλήρης και εύχρηστος διαχειριστής αρχείων, σε θέση να ενσωματωθεί στο GNOME.

Αυτό σημαίνει ότι η ιστορία της Eazel και του Ναυτίλου μπορεί να θεωρηθεί σαν μία παραδειγματική περίπτωση ενός προγράμματος που επέζησε ακόμη και μετά την εξαφάνιση της εταιρείας που το δημιούργησε. Αυτό είναι κάτι που είναι εφικτό μόνο στον κόσμο του Ελεύθερου Λογισμικού.

- Η Red Hat ίδρυσε την Red Hat Advanced Development Labs, RHAD, με σκοπό να εξασφαλίσει ότι η επιφάνεια εργασίας του GNOME θα κέρδιζε σε φιλικότητα προς τον χρήστη και σε ισχύ. Για να το επιτύχει, η Red Hat απασχόλησε πέντε-έξι από τους σημαντικότερους *hackers* του GNOME και τους έδωσε την ελευθερία να αναπτύξουν οτιδήποτε θεωρούσαν απαραίτητο. Από την RHAD Labs έχουμε το ORBit, την υλοποίηση του CORBA, που χρησιμοποιήθηκε στο GNOME project, και ήταν γνωστό σαν το "γρηγορότερο πιστόλι στην Δύση". Ακόμη μία σημαντική πτυχή είναι το έργο που έγινε στη νέα έκδοση της GTK+ και στο σύστημα ρύθμισης του GNOME, το GConf.

- Η SUN Microsystems υπεισήλθε στην ανάπτυξη του GNOME σε ένα πιο ώσιμο στάδιο, καθώς το GNOME είχε πλέον γίνει ένα σχετικά ώριμο προϊόν, μέχρι το Σεπτέμβριο του 2000. Η πρόθεση της SUN ήταν να χρησιμοποιήσει το GNOME ως γραφικό περιβάλλον για την επιφάνεια εργασίας του λειτουργικού συστήματος Solaris. Δημιούργησε, λοιπόν, μία ομάδα που θα εργαζόταν πάνω στο GNOME, του οποίου τα πιο δυνατά σημεία περιλαμβάνουν την ευχρηστία και την εύκολη προσβασιμότητα. Τον Ιούνιο του 2003, η SUN ανακοίνωσε ότι θα διένειμε το GNOME 2.2 μαζί με την έκδοση 9 του Solaris.

9.4.4. Η τρέχουσα κατάσταση του GNOME

Το GNOME, από τις αρχές του 2007, βρίσκεται στη έκδοση 2.18. Οι περισσότερες από τις τεχνολογίες στις οποίες βασίζεται έχουν ωριμάσει, όπως είναι προφανές και από την αρίθμηση της έκδοσης. Για παράδειγμα, ο διαμεσολαβητής CORBA που χρησιμοποιείται τώρα, είναι το ORBit2, ενώ το γραφικό περιβάλλον καθώς και το API, το GTK+, υπέστησαν τροποποιήσεις που προέκυψαν από την εμπειρία που συσσωρεύθηκε από τις προηγούμενες εκδόσεις του GNOME.

Μία σημαντική καινοτομία είναι η προσθήκη μίας βιβλιοθήκης προσβασιμότητας, που προτάθηκε από την SUN, και η οποία επιτρέπει στους ανθρώπους με ειδικές ανάγκες να χρησιμοποιούν το περιβάλλον GNOME. Ιδιαίτερη μνεία θα πρέπει να γίνει και για το Bonobo, το σύστημα των επί μέρους συστατικών του GNOME. Το Bonobo άφησε το σημάδι του ενόσω ήταν μέσα στο GNOME, και ενώ αναπτυσσόταν το Evolution, ένα πρόγραμμα διαχείρισης προσωπικών πληροφοριών. Συν τω χρόνω, όμως, απεδείχθη ότι οι προσδοκίες που είχε δημιουργήσει το Bonobo ήταν υπέρμετρες και ότι η επαναχρησιμοποίηση της προσπάθειας που επενδύθηκε σε αυτό, με τη χρήση των συστατικών του, δεν ήταν τόσο εξαντλητική όσο αναμενόταν αρχικά.

Σημείωση

Η βιβλιοθήκη ATK είναι μία βιβλιοθήκη με αφηρημένες κλάσεις (*abstract classes*), που κάνει τις εφαρμογές προσβάσιμες. Αυτό σημαίνει ότι ορισμένοι άνθρωποι με ειδικές ανάγκες (με προβλήματα όρασης ή αχρωματοψία, όσοι δεν μπορούν να χρησιμοποιήσουν το ποντίκι, ή ένα πληκτρολόγιο, κλπ.) μπορούν, παρόλα ταύτα, να χρησιμοποιούν το GNOME. Το ενδιαφέρον της SUN να διασφαλίσει την προσβασιμότητα, οφείλεται στο γεγονός πως εάν επιθυμεί να προσφέρει τις υπηρεσίες της προς την ομοσπονδιακή κυβέρνηση των ΗΠΑ, τότε θα πρέπει να πληροί μία σειρά προδιαγραφών σχετικών με την προσβασιμότητα. Η SUN πήρε αυτή τη δουλειά τόσο στα σοβαρά, που προσέλαβαν και έναν τυφλό προγραμματιστή στην ομάδα εργασίας για την ανάπτυξη του GNOME. Τον Σεπτέμβριο του 2002, η αρχιτεκτονική προσβασιμότητας του GNOME κέρδισε το βραβείο Helen Keller Achievement Award.

9.4.5. Ακτινογραφία του GNOME

Τα δεδομένα και οι αριθμοί του πίνακα 10, μας φέρνουν στο τέλος της παρουσίασης του GNOME. Οι αριθμοί αυτοί αποτυπώνουν την κατάσταση στο CVS του GNOME, όπως είχε στις 14/08/2003. Σε εκείνη την ημερομηνία, υπήρχαν περισσότερες από 9.000.000 γραμμές κώδικα που φιλοξενούνταν στο αποθετήριο του CVS, ιδιοκτησίας του GNOME project. Έστω και αν το πιο φυσικό πράγμα θα ήταν να συγκρίνουμε το GNOME με το KDE, πρέπει να προειδοποιήσουμε τους αναγνώστες πως οι υπάρχουσες διαφορές, από την πλευρά του τρόπου οργάνωσης αυτών των projects, καθιστούν μία τέτοια σύγκριση άσκοπη, αν μας ενδιαφέρει συγκρίνουμε όμοια με όμοια, “επ’ ίσοις όροις”. Αυτό συμβαίνει, για παράδειγμα, διότι το CVS του GNOME περιλαμβάνει το GIMP (ένα πρόγραμμα για την δημιουργία και την διαχείριση γραφικών), το οποίο, από μόνο του, αντιπροσωπεύει περισσότερες από 660.000 γραμμές κώδικα, ή την βιβλιοθήκη GTK+, πάνω στην οποία εστιάζεται η ανάπτυξη του GNOME, και η οποία, από μόνη της, έχει 330.000 γραμμές. Αν, πέραν αυτού, προσθέσουμε το γεγονός ότι το αποθετήριο CVS του GNOME τείνει περισσότερο να ανοίγει νέες ενότητες/modules για τα προγράμματα (διαθέτει, συνολικά, 700) από ότι εκείνο του KDE (που έχει λιγότερα από 100), τότε μπορούμε να κατανοήσουμε γιατί το GNOME έχει περισσότερες γραμμές από το KDE, παρά το ότι είναι νεότερο κατά ενάμιση χρόνο. Το αποθετήριο του GNOME φιλοξενεί παραπάνω από 225.000 αρχεία, που προσετέθησαν και τροποποιήθηκαν σχεδόν 2.000.000 φορές (βλέπε τον αριθμό των *commits* λίγο παρακάτω, στον πίνακα).

Σημείωση

Μία εταιρεία που θα ήθελε να φτιάξει λογισμικό στο μέγεθος του GNOME, θα έπρεπε να προσλάβει, κατά μέσον όρο, περίπου 250 developers, επί περισσότερο από 11 χρόνια, για να έχει ένα προϊόν με παρόμοιες δυνατότητες, σύμφωνα με το μοντέλο COCOMO που χρησιμοποιήθηκε σε όλο αυτό το κεφάλαιο. Το σχετικό κόστος θα ανερχόταν περίπου στα 400.000.000 δολάρια, ένα ποσό παρόμοιο με εκείνο που επένδυσε μία επιτυχημένη εταιρεία κινητής τηλεφωνίας το 2003, για να ενισχύσει τη χωρητικότητα του δικτύου της, ή παρόμοιο με το ποσό που μία αυτοκινητοβιομηχανία θα έπρεπε να επενδύσει για να ανοίξει παραγωγικές εργοστασιακές εγκαταστάσεις στη Βαρκελώνη.

Στους ανθρώπινους πόρους του GNOME περιλαμβάνονται σχεδόν 1000 developers με δικαιώματα write-access στο σύστημα ελέγχου αναθεώρησης των εκδόσεων CVS, εκ των οποίων σχεδόν 20 εργάζονται επαγγελματικά για το GNOME (είτε full-time, είτε part-time). Μόνον το 25% από αυτούς ήταν δραστήριοι κατά τον τελευταίο χρόνο, ενώ το 40% ήταν δραστήριοι κατά τα τελευταία δύο χρόνια. Ο μέσος αριθμός *commits* ανά ημέρα, που κατεγράφησαν από τότε που το project ιδρύθηκε, είναι σχεδόν 1000. Τα εργαλεία για την υποστήριξη της ανάπτυξης που χρησιμοποιούνται από το GNOME project είναι, βασικά, τα ίδια με εκείνα σε χρήση από το KDE, και γι' αυτό δεν θα τα εξετάσουμε πολύ εις βάθος.

Πίνακας 10. Ανάλυση του GNOME

Ιστότοπος	http://www.gnome.org
Έναρξη του project	Σεπτέμβριος 1997
Άδεια	GNU GPL και GNU LPGL
Έκδοση που αναλύθηκε	2.2
Γραμμές πηγαίου κώδικα.	9.200.000
Αριθμός αρχείων (κώδικας, τεκμηρίωση, κλπ.)	228.000
Εκτίμηση κόστους	\$ 400.000.000
Εκτίμηση απαιτούμενου χρόνου ως την ολοκλήρωση	11.08 έτη (133.02 μήνες)
Εκτίμηση μέσου αριθμού developers	Περίπου 250

Αριθμός subprojects	Πάνω από 700 modules στο CVS.
Κατ' εκτίμηση αριθμός των developers	Περί τους 1000 με δικαίωμα write-access στο CVS.
Αριθμός ενεργών <i>committers</i> κατά τον τελευταίο χρόνο	Περί τους 500 (περίπου το 55% του συνόλου)
Αριθμός ενεργών <i>committers</i> κατά τα τελευταία 2 χρόνια	Περί τους 700 (75% του συνόλου)
Αριθμός <i>commits</i> στο CVS	1.900.000
Μέσος αριθμός <i>commits</i> (σύνολο) ανά ημέρα	Περίπου 900
Εργαλεία υποστήριξης ανάπτυξης	CVS, mailing lists, ιστοσελίδα, news site, ετήσιες συναντήσεις

Ενώ στο KDE, η πιο πολυχρησιμοποιημένη γλώσσα είναι αναμφισβήτητα η C++, στο GNOME, η κυρίαρχη γλώσσα είναι η C. Στο GNOME, όπως συμβαίνει και στο KDE, αυτό οφείλεται στο γεγονός ότι η βασική βιβλιοθήκη είναι γραμμένη σε C, οπότε και η ενδογενής (native) γλώσσα είναι η C, ενώ οι προγραμματιστές που θέλουν να χρησιμοποιήσουν άλλες γλώσσες, θα πρέπει να περιμένουν να εμφανισθούν τα αντίστοιχα bindings. Το πιο προχωρημένο language binding στο περιβάλλον GNOME είναι εκείνο που ενσωματώνεται στο Gnome--, που δεν είναι άλλο από την C++. Γι' αυτό, δεν αποτελεί έκπληξη που αυτή είναι η 2^η γλώσσα στην κατάταξη. Η Perl είχε ανέκαθεν γίνει ευρέως αποδεκτή από την κοινότητα GNOME και ένα παράδειγμα αυτού του γεγονότος είναι ότι στο GNOME είναι δυνατός ο προγραμματισμός σε πολλές γλώσσες. Η υλοποίησή της, πάντως, δεν ήταν τόσο εκτεταμένη όσο θα περίμενε κανείς και είναι ελαφρώς πιο εκτεταμένη από εκείνη του Shell. Σε ένα άλλο επίπεδο, οι γλώσσες Python και Lisp χρησιμοποιήθηκαν σε αρκετά μεγάλη έκταση στο GNOME, όπως αποδεικνύεται από τη σχετική σημαντικότητα αυτής της κατάταξης, ενώ η Java δεν μπόρεσε ποτέ να ριζώσει, ίσως λόγω ενός ατελούς συνδέσμου.

Πίνακας 11. Προγραμματιστικές γλώσσες που χρησιμοποιούνται στο GNOME

Προγραμματιστική γλώσσα	Γραμμές Κώδικα	Ποσοστό
C	7.918.586	86.10%
C++	576.869	6.27%
Perl	199.448	2.17%
Shell	159..63	1.73%
Python	137,380	1.49%
Lisp	88.546	0.96%

9.4.6. Ακαδημαϊκές μελέτες στο GNOME

Οι πιο σημαντικές ακαδημαϊκές μελέτες στο GNOME, είναι οι εξής δύο: "Results from software engineering research into open source development projects using public data" [158] και το "The evolution of GNOME" [123].

- Η [158] αποτελεί μία από τις πρώτες, μεγάλης κλίμακος, μελέτες λογισμικού στο χώρο του Ελεύθερου Λογισμικού. Οι συγγραφείς αυτής της μελέτης εκμεταλλεύθηκαν το γεγονός ότι οι

λεπτομέρειες της ανάπτυξης είναι συνήθως δημόσια διαθέσιμες, για να μπορούν να μετρώνται τα κόστη και οι χρόνοι ανάπτυξης και να συγκρίνονται με τις προβλέψεις των μοντέλων εκτίμησης κόστους, καθώς και τις παραδοσιακές μετρήσεις χρόνου και ανθρώπινης προσπάθειας. Ένα από τα κλασικά μοντέλα με τα οποία συγκρίθηκαν, ήταν εκείνο που χρησιμοποιήθηκε σε αυτό το κεφάλαιο, το μοντέλο COCOMO.

- Η [123], εν συντομία, ανασκοπεί τους στόχους του GNOME και τη σύντομη ιστορία του, καθώς και τη χρήση τεχνολογίας, στο πλαίσιο του GNOME project.

9.5. Το Apache

Ο Διακομιστής HTTP Apache, είναι μία από τις κορυφαίες εφαρμογές του κόσμου του Ελεύθερου Λογισμικού, καθώς αποτελεί έναν διακομιστή διαδικτύου (*Web server*) που χρησιμοποιείται πολύ διάχυτα, σύμφωνα με την έρευνα πραγματικού χρόνου της Netcraft

(http://news.netcraft.com/archives/2003/08/01/august_2003_web_server_survey.html) [167]. Για παράδειγμα, τον Μάιο του 1999, το 57% των διακομιστών του διαδικτύου, δούλευαν με Apache, ενώ το Μάιο του 2003, το ποσοστό αυτό αυξήθηκε στο 68%. Το Apache διατίθεται για όλους τα είδη συστημάτων Unix (BSD, GNU/Linux, Solaris...), για τα MS Windows και για άλλες δευτερεύουσες πλατφόρμες.

9.5.1. Η ιστορία του Apache

Τον Μάρτιο του 1989, ο Tim Berners Lee, ένας Άγγλος επιστήμονας που δούλευε στο CERN (Ελβετία), πρότεινε μία νέα μέθοδο για την διαχείριση των τεραστίων ποσοτήτων πληροφοριών που διακινούνται στα CERN projects. Η μέθοδος θα βασιζόταν σε ένα δίκτυο κειμένων διασυνδεδεμένων με υπερσυνδέσμους (*hyperlinks*). Ο Ted Nelson, ήδη από το 1965, το ονόμασε υπερκείμενο (“hypertext”): γεννήθηκε, έτσι, το WWW. Δεν ήταν, όμως, παρά τον Νοέμβριο του 1990 που εμφανίσθηκε το πρώτο λογισμικό WWW: ένα πακέτο που ονομαζόταν “World Wide Web” και που περιελάμβανε ένα περιηγητή διαδικτύου (*web browser*) με μία γραφική διεπαφή και έναν επεξεργαστή WYSIWYG “what you see is what you get” (“ό,τι βλέπεις, εκείνο παίρνεις”). Δύο χρόνια αργότερα, ο κατάλογος των διακομιστών WWW περιείχε περίπου 30 καταχωρήσεις, περιλαμβανομένης της NCSA HTTPd.

Η πραγματική ιστορία του Apache άρχισε όταν ο Rob McCool άφησε την NCSA τον Μάρτιο του 1995. Θα γεννιόταν ο Apache 0.2, στις 18/03/1995, βασιζόμενος στον διακομιστή NCSA HTTPd 1.3 server, που φτιάχθηκε από τον ίδιον τον Rob McCool, ενόσω ήτανε στην NCSA. Κατά τη διάρκεια αυτών των πρώτων μηνών, ο Apache απετελείτο από μία συλλογή patches που εφαρμόστηκαν στον διακομιστή NCSA, μέχρις ότου ο Robert Thau λανσάρισε τον Shambhala 0.1, μία σχεδόν εκ νέου επαναυλοποίηση, που ήδη εμπεριείχε τα API για τις ενότητες/modules που, ακολούθως, απέβησαν τόσο πετυχημένα.

Σημείωση

Το όνομα του Apache project βασίζεται στη φιλοσοφία του σχετικά με την ανάπτυξη και την οργάνωση. Όπως συνέβαινε συνήθως στην φυλή των Apache, οι developers του Apache αποφάσισαν ότι η οργανωτική τους μέθοδος θα πρέπει να βασίζεται στις προσωπικές αξίες των developers, συγκριτικά με την υπόλοιπη κοινότητα Apache. Υπάρχει, πάντως, ένας διαδεδομένος μύθος που λέει ότι το όνομα Apache προέρχεται, στην πραγματικότητα, από το ότι, στα αρχικά στάδια, επρόκειτο απλώς για έναν “πατσarisμένο” διακομιστή NCSA, δηλ. έναν patchy (κηλιδωτό και ετερόκλητο) διακομιστή.

Η πρώτη σταθερή έκδοση του Apache δεν εμφανίσθηκε ως τον Ιανουάριο του 1996, όταν κυκλοφόρησε ο Apache 1.0, που περιελάμβανε την φόρτωση αρθρωμάτων σε test-mode, κατά την

εκτέλεση του προγράμματος, όπως και άλλες ενδιαφέρουσες λειτουργίες. Οι πρώτοι μήνες εκείνου του έτους υπήρξαν ιδιαίτερα παραγωγικοί για το project, καθώς η έκδοση 1.1, που διέθετε λειτουργίες ταυτοποίησης χρήστη (*authentication modules*), οι οποίες θα ελέγχονταν σε συνδυασμό με τις βάσεις δεδομένων (σαν την MySQL), δημοσιοποιήθηκε μόνο 2 μήνες αργότερα. Από τότε μέχρι και σήμερα, τα πιο σημαντικά γεγονότα για το project ήταν η εισαγωγή της πλήρους συμβατότητας με τις προδιαγραφές του HTTP 1.1 (ενσωματώθηκε στον Apache 1.2 τον Απρίλιο του 1997), η συμπερίληψη της πλατφόρμας Windows NT (άρχισε τον Ιούλιο του 1997 με τις δοκιμαστικές εκδόσεις του Apache 1.3), η ενοποίηση των αρχείων ρύθμισης σε ένα και μοναδικό αρχείο (που δεν ενσωματώθηκε παρά στον Apache 1.3.3, τον Οκτώβριο του 1998) και το λανσάρισμα, σε δοκιμαστικό ακόμη στάδιο, της νέας γενιάς του Apache, του Apache 2.

Στο εν τω μεταξύ, τον Ιούνιο του 1998, η IBM αποφάσισε ότι, αντί να αναπτύσσει το δικό της HTTP, θα χρησιμοποιούσε τον Apache ως μηχανή, για το δικό της προϊόν WebSphere. Αυτό ερμηνεύθηκε σαν μία τεράστια επιδοκμασία για το Apache project, αλλά και, γενικά, για το Ελεύθερο Λογισμικό, από την εταιρεία Big Blue (το άλλο όνομα για την IBM), αν και έπρεπε να τροποποιηθεί ελαφρώς η αρχική Άδεια Apache, για να μπορεί αυτό το σχήμα να δουλέψει.

9.5.2. Η ανάπτυξη του Apache

Ο διακομιστής HTTP Apache αποτελεί το κύριο project, ανάμεσα στα τόσα άλλα, που διαχειρίζεται το Apache Software Foundation. Ο αρθρωτός σχεδιασμός του Apache έκανε εφικτή την ύπαρξη μίας σειράς δορυφορικών projects, βασισμένων στο Apache, μερικά από τα οποία εξελίχθηκαν και έγιναν μεγαλύτερα ακόμη και από το ίδιο το Apache. Στο διακομιστή HTTP Apache, π.χ., ο διακομιστής περιέχει τον πυρήνα του συστήματος με τις βασικές λειτουργίες, ενώ οι πρόσθετες λειτουργίες παρέχονται από διαφορετικά modules. Τα πιο γνωστά *modules* είναι το *mod_perl* (ένας διερμηνέας για scripts της γλώσσας Perl, που είναι ενσωματωμένος στον web server) και η *Jakarta* (ένας ισχυρός διακομιστής εφαρμογών). Στις επόμενες παραγράφους, θα περιγράψουμε μόνο την διαδικασία ανάπτυξης που ακολουθείται για τον διακομιστή HTTP, χωρίς να λάβουμε υπ' όψη μας τα άλλα modules, τα οποία μπορεί να διαθέτουν αντίστοιχες διαδικασίες, ή όχι.

Η ανάπτυξη του διακομιστή HTTP Apache βασίζεται στην δουλειά μίας μικρής ομάδας *developers*, γνωστής με το όνομα Apache Group. Το Apache Group αποτελείται από *developers* που δούλεψαν μαζί στο project για ένα μεγάλο διάστημα, γενικά, πάνω από 6 μήνες. Ένας τέτοιος *developer*, όταν παίρνει μία πρόσκληση να συμμετάσχει από ένα εκ των μελών του Apache Group, ψηφίζεται από όλα τα υπόλοιπα μέλη. Στα αρχικά στάδια, το Apache Group αποτελείτο από οκτώ *developers*. Αυτός ο αριθμός αυξήθηκε, στη συνέχεια, στους 12, ενώ τώρα υπάρχουν 25 μέλη.

Το Apache Group είναι υπεύθυνο για την ανάπτυξη του web server και, άρα, για συγκεκριμένες αποφάσεις, που αφορούν την ανάπτυξη ανά οιαδήποτε χρονική στιγμή. Είναι σημαντικό να διαχωρίζουμε το Apache Group από τους *developers* εντός της *Κεντρικής Ομάδας*, που είναι ενεργή σε όλες τις περιστάσεις. Η εθελοντική φύση της εργασίας που προσφέρουν οι περισσότεροι *developers*, καθιστά απίθανο να μπορούν όλοι οι άνθρωποι που αποτελούν το Apache Group να είναι ενεργοί ανά πάσα στιγμή, πράγμα που σημαίνει ότι ως πυρήνας ορίζεται εκείνη η ομάδα ανθρώπων που επιμελούνται τις διάφορες εργασίες στο Apache, για μία δεδομένη χρονική περίοδο. Σε γενικές γραμμές, οι αποφάσεις που πρέπει να παίρνουν οι *developers* που ανήκουν στην ομάδα του πυρήνα, περιορίζονται στην ψηφοφορία υπέρ του να συμπεριληφθεί ή όχι κάποιο κομμάτι κώδικα, αν και στην πραγματικότητα αυτό επιφυλάσσεται μόνο για τις αλλαγές μεγάλης κλίμακος και για θέματα σχεδιασμού. Από την άλλη πλευρά, έχουν συνήθως δικαιώματα *write-access* στο αποθετήριο CVS, που σημαίνει ότι ενεργούν ως φύλακες έναντι του εισερχόμενου νέου κώδικα, εξασφαλίζοντας ότι πρόκειται για σωστό και καλής ποιότητας κώδικα.

9.5.3. Ακτινογραφία του Apache

Οι αριθμοί που εμφανίζονται πιο κάτω αντιστοιχούν στην έκδοση του διακομιστή HTTP Apache που ήταν διαθέσιμη για κατέβασμα από τον διακομιστή του CVS για το Apache project, στις 18 Απριλίου του 2003. Κανένα από τα πολυάριθμα modules που διαθέτει το Apache project δεν ελήφθησαν υπ' όψιν εδώ. Όπως θα δούμε, το Apache project είναι σχετικά μικρό όταν το συγκρίνουμε με τις άλλες περιπτώσεις που μελετήσαμε σε αυτό το κεφάλαιο. Αν και αυτό το έχουμε ήδη αναφέρει, είναι σημαντικό να δώσουμε έμφαση στην διαμορφωσιμότητα (modularity) του Apache, που παρουσιάζει τα εξής ειδικά πλεονεκτήματα: ο πυρήνας είναι μικρός και διαχειρίσιμος. Το αποθετήριο CVS του Apache project, που περιέχει τον πυρήνα του web server και πολλά πρόσθετα modules, φιλοξενεί περισσότερες από τέσσερα εκατομμύρια (4.000.000) γραμμές πηγαίου κώδικα, ένας αριθμός που είναι ελαφρώς κατώτερος από εκείνους σε projects σαν το KDE και το GNOME.

Η έκδοση 1.3 του Apache είχε λίγο παραπάνω από 85.000 γραμμές πηγαίου κώδικα. Σύμφωνα με το μοντέλο COCOMO, αυτό θα απαιτούσε την δουλειά, κατά μέσον όρον, είκοσι (20) developers, που θα έπρεπε να εργάζονται full-time επί ενάμισι (1.5) έτος. Το συνολικό κόστος του project θα ήταν, εκείνη την εποχή, περίπου τέσσερα εκατομμύρια (4.000.000) δολάρια. Για να ολοκληρώσουν τον Apache web server, θα ήταν απαραίτητοι έως και εξήντα (60) διαφορετικοί *committers*, ενώ ο αριθμός των developers που θα έπρεπε να παρέχουν input, σύμφωνα με τους υπολογισμούς, θα έπρεπε να ήταν περίπου τετρακόσιοι (400).

Πίνακας 12. Ανάλυση του Apache

Ιστότοπος	http://www.apache.org
Έναρξη του project	1995
Άδεια	Apache Free Software License
Έκδοση που αναλύθηκε	2.2.4
Γραμμές Πηγαίου Κώδικα.	225.065
Αριθμός αρχείων	2.807
Εκτίμηση κόστους	\$ 7.971.958
Εκτίμηση απαιτούμενου χρόνου ως την ολοκλήρωση	2.52 έτη (30.27 μήνες)
Εκτίμηση μέσου αριθμού developers	23.4
Εκτιμώμενος αριθμός developers	60 committers (400 developers)
Εργαλεία υποστήριξης ανάπτυξης	CVS, mailing lists, σύστημα αναφοράς των bugs

Ο Apache 1.3 είναι γραμμένος σχεδόν εξ' ολοκλήρου σε γλώσσα C και σπανίως υπάρχουν οποιεσδήποτε άλλες προγραμματιστικές γλώσσες, ειδικά αν λάβουμε υπ' όψη μας ότι οι περισσότερες γραμμές που γράφτηκαν στην δεύτερη γλώσσα, την Shell, αντιστοιχούν σε αρχεία ρύθμισης και υποβοήθησης της μεταγλώττισης.

Πίνακας 13. Προγραμματιστικές γλώσσες που χρησιμοποιούνται στο Apache

Γλώσσα Προγραμματισμού	Γραμμές Κώδικα	Ποσοστό
C	208.866	92.8%
Shell	12.796	5.69%
Perl	1.649	0.73%
Awk	874	0.39%

9.6. To Mozilla

Το Mozilla project δουλεύει με ένα σύνολο ολοκληρωμένων εφαρμογών για το Διαδίκτυο, που είναι ελεύθερες και διαπλατφορμικές, και τα πιο αξιοσημείωτα προϊόντα είναι ο περιηγητής διαδικτύου Mozilla Firefox και ο Mozilla Thunderbird πελάτης για e-mail και news. Και αυτή η ομάδα έχει σχεδιασθεί σαν μία πλατφόρμα που να επιτρέπει την ανάπτυξη άλλων εφαρμογών, το οποίο σημαίνει πως υπάρχουν πολλοί άλλοι περιηγητές διαδικτύου (σαν το Galeon) που χρησιμοποιούν επίσης το Gecko, τη μηχανή HTML του Mozilla.

Η διαχείριση του προγράμματος γίνεται από το Ίδρυμα Mozilla Foundation, μία οργάνωση Μ.Κ.Ο. που φτιάχνει ελεύθερο λογισμικό και που είναι "αφιερωμένη στη διατήρηση της δυνατότητας επιλογής και στην προώθηση της καινοτομίας στο Διαδίκτυο". Για τον λόγο αυτό, τα προϊόντα του Mozilla βασίζονται σε τρεις βασικές αρχές: πρέπει να είναι Ελεύθερο Λογισμικό, να σέβονται τις προδιαγραφές και να είναι φορητά και σε άλλες πλατφόρμες.

9.6.1. Ιστορία του Mozilla

Η ιστορία του Mozilla είναι μεγάλη και μπερδεμένη αλλά και πολύ ενδιαφέρουσα, καθώς μας δίνει την δυνατότητα να παρακολουθήσουμε την ιστορία του ίδιου του WWW. Αυτό συμβαίνει διότι αν ανατρέξουμε σε όλους εκείνους τους ανθρώπους και τους οργανισμούς που ενεπλάκησαν στην ανάπτυξη του Mozilla, τότε θα φθάσουμε στο εναρκτήριο σημείο του Διαδικτύου, με το λανσάρισμα του πρώτου πλήρους περιηγητή Διαδικτύου.

Όπως συνέβη και με την περίπτωση του προκατόχου του Apache, το NCSA ήταν το μέρος όπου "γεννήθηκε" ο Mosaic, ο πρώτος πλήρης περιηγητής Διαδικτύου, το 1993. Πολλά από τα μέλη της ομάδας ανάπτυξης, με τους Marc Andreessen και Jim Clark επικεφαλής, έφτιαξαν μία μικρή εταιρεία με σκοπό να γράψουν, αρχίζοντας εκ του μηδενός (καθώς υπήρχαν προβλήματα με τα πνευματικά δικαιώματα/copyright του κώδικα του Mosaic και τον τεχνικό σχεδιασμό του προγράμματος που είχε τα όριά του, βλέπε "*Speeding the Net: the inside story of Netscape and how it challenged Microsoft*" [189]), αυτό που στη συνέχεια θα γινόταν ο περιηγητής Netscape Communicator, ο οποίος ήταν, αναμφισβήτητα, ο ηγέτης της αγοράς των περιηγητών Διαδικτύου μέχρι την άφιξη του Microsoft Internet Explorer. Εκτός από την καθαρά τεχνολογική καινοτομία που αντιπροσώπευε ο Netscape, η εταιρεία Netscape Inc. αποδείχθηκε, επίσης, καινοτόμος στον τρόπο με τον οποίο κατάφερε να στριμώξει την αγορά. Εντελώς αντίθετα με ότι εθεωρείτο ως "κοινή λογική", εκείνη την εποχή, η κορυφαία της εφαρμογή, ο περιηγητής WWW, προσφερόταν δωρεάν (και μπορούσε ακόμη και να διανέμεται, με κάποιους περιορισμούς). Αυτή η προσέγγιση, που ήταν απολύτως ανήκουστη στον εταιρικό κόσμο, τον καιρό εκείνο, προκάλεσε κάποια έκπληξη, αλλά τελικά αποδείχθηκε ότι ήταν σωστή για την στρατηγική της Netscape Inc., και μόνον ο γίγαντας με το όνομα Microsoft ήταν ικανός να το εκτοπίσει με περισσότερες επιθετικές τακτικές.

Γύρω στο 1997, το μερίδιο αγοράς του Netscape είχε πέσει δραστικά, λόγω της εξάπλωσης του Microsoft Explorer; οπότε η Netscape Inc. μελετούσε νέους τρόπους για την ανάκτηση της προηγούμενης κυριαρχίας της. Μία τεχνική αναφορά που δημοσίευσε ο μηχανικός Frank Hecker ("*Setting up shop: the business of open source software*", 1998) [142] πρότεινε σαν καλύτερη λύση για το πρόβλημα αυτό να απελευθερωθεί (release) ο πηγαίος κώδικας του περιηγητή, ώστε να δρέψει τα οφέλη της κοινότητας του Ελεύθερου Λογισμικού, όπως περιέγραφε ο Eric Raymond στο "The Cathedral and the Bazaar".

Τον Ιανουάριο του 1998, η Netscape Inc. ανακοίνωσε επισήμως ότι θα έθετε τον πηγαίο κώδικα του περιηγητή της σε δημόσια διάθεση, θέτοντας έτσι έναν εξαιρετικά σημαντικό θεμέλιο λίθο μέσα στη σύντομη ιστορία του Ελεύθερου Λογισμικού: μία εταιρεία θα δημοσιοποιούσε όλο τον πηγαίο κώδικα μίας εφαρμογής, που μέχρι τότε ήταν ένα εμπορικό προϊόν, υπό μία Άδεια Ελεύθερου Λογισμικού. Η ημερομηνία για το λανσάρισμα προγραμματίστηκε για τις 31 Μαρτίου του 1998.

Ανάμεσα στους δύο μήνες, τον Ιανουάριο και τον Μάρτιο, οι άνθρωποι του Netscape ήταν σε φρενήρη δραστηριότητα, προσπαθώντας να ετοιμάσουν τα πάντα. Ο κατάλογος των εργασιών ήταν τεράστιος και περίπλοκος ("*Freeing the source: the story of Mozilla*", 1999) [134]. Στο τεχνικό επίπεδο, ήταν απαραίτητο να έρθουν σε επαφή με τις εταιρείες που είχαν φτιάξει τα modules, για να τους ζητήσουν τη συναίνεσή τους για να αλλάξουν την Άδεια. Αν η απάντηση ήταν αρνητική, τότε το module θα έπρεπε να απομακρυνθεί. Επιπρόσθετα, όλα τα μέρη που γράφτηκαν σε Java θα έπρεπε να επαναυλοποιηθούν, καθώς θεωρήθηκε ότι η Java δεν ήταν ελεύθερη. Κατόπιν, αποφάσισαν να ονομάσουν το ελεύθερο αυτό project Mozilla, ακριβώς με τον ίδιο τρόπο με τον οποίο οι developers του Netscape είχαν ονομάσει το βασικό τους συστατικό Mozilla. Επίσης, αγοράστηκε και το domain Mozilla.org για να χτισθεί μία κοινότητα developers και βοηθών, που θα βασιζόταν πάνω σε αυτόν τον ιστότοπο. Στο τέλος της διαδικασίας, αποδόθηκαν παραπάνω από ενάμισι εκατομμύριο (1.500.000) γραμμών πηγαίου κώδικα.

Σημείωση

Το όνομα Mozilla είναι ένα λογοπαίγνιο, με μία μικρή δόση χιούμορ από την ομάδα ανάπτυξης του Netscape Inc.. Το όνομα Mozilla προήλθε από την προσαρμογή του ονόματος Godzilla (το τέρας που προκαλούσε καταστροφή στις Ιαπωνικές ταινίες τρόμου κατά την δεκαετία του 1950), για να το κάνουν να ακούγεται σαν *Mosaic Killer*, καθώς ο νέος browser, με την πιο προηγμένη τεχνολογία, υποτίθεται πως θα μετέτρεπε το Mosaic σε απαρχαιωμένο.

Σε ένα άλλο επίπεδο, υπήρχε ένα νομικό θέμα. Οι ελεύθερες άδειες που υπήρχαν την εποχή εκείνη δεν έπεισαν τα διοικητικά στελέχη του Netscape, που δεν έβλεπαν πως αυτές θα μπορούσαν να είναι "συμβατές" με την εμπορική φύση μίας επιχείρησης. Το Netscape ήθελε μία πιο *ενέλικτη* άδεια, που να καθιστά εφικτές τις συμφωνίες με τρίτους, για να μπορεί να συμπεριλάβει τον κώδικά τους, ανεξαρτήτως του τύπου της Αδείας αλλά και ασχέτως από τα εάν άλλοι εμπορικοί developers επρόκειτο να συνεισφέρουν σε αυτό, και έτσι, να μπορούν να υπερασπισθούν τα οικονομικά τους συμφέροντα με οποιονδήποτε τρόπο θα επέλεγαν. Αν και δεν είχαν αρχικά σχεδιάσει την δημιουργία μίας νέας Αδείας, κατέληξαν ενδεχομένως στο συμπέρασμα ότι αυτός ήταν ο μόνος τρόπος με τον οποίο θα μπορούσαν να πετύχουν αυτό που ήθελαν. Αυτός είναι ο τρόπος με τον οποίο δημιουργήθηκε η Άδεια Netscape Public License (NPL): μία άδεια που στηριζόταν στις βασικές αρχές των αδειών ελεύθερου λογισμικού, αλλά που παραχωρούσε ορισμένα πρόσθετα δικαιώματα προς την Netscape Inc, πράγμα που την καθιστούσε μία *μη-ελεύθερη* άδεια, από την σκοπιά του Free Software Foundation. Όταν δημοσιεύθηκε το προσχέδιο της NPL και ετέθη σε δημόσια συζήτηση, εκείνη η συγκεκριμένη ρήτρα, που παραχωρούσε πρόσθετα δικαιώματα στη Netscape, συγκέντρωσε έντονη κριτική. Σε ανταπάντηση προς αυτές τις κριτικές, η Netscape Inc. αντέδρασε γρήγορα και δημιούργησε μία επιπλέον άδεια, τη Mozilla Public License (MPL), η οποία ήταν ίδια με την NPL, εκτός από το ότι η Netscape δεν είχε πλέον πρόσθετα δικαιώματα.

Η τελική απόφαση ήταν να αποδοθεί ο κώδικας του Netscape υπό την άδεια NPL, που παρείχε πρόσθετα δικαιώματα στην Netscape, αλλά οποιοσδήποτε νέος κώδικας που θα ενσωματωνόταν, θα υπαγόταν στην άδεια MPL (ή σε κάποια συμβατή με αυτήν άδεια). Οι διορθώσεις στον αρχικό κώδικα (που κυκλοφορούσε υπό την NPL), θα καλυπτόταν επίσης από αυτή την άδεια.

Σημείωση

Προς το παρόν, το Mozilla δέχεται συνεισφορές υπό τρεις άδειες: την MPL, την GPL και την LGPL. Η αλλαγή της άδειας δεν ήταν καθόλου εύκολη, καθώς έπρεπε να βρεθούν όλοι εκείνοι οι άνθρωποι που είχαν συνεισφέρει κώδικα σε οποιοδήποτε σημείο, για να δώσουν την συναίνεσή τους σε αυτή την μετάβαση από το σχήμα NPL/MPL στον συνδυασμό MPL/GPL/LGPL. Για γίνει εφικτή η εκ νέου αδειοδότηση ολόκληρου του κώδικα, στήθηκε ένας ιστότοπος, ο οποίος περιείχε έναν κατάλογο 300 "απωλεσθέντων" hackers (*"Have you seen these hackers?"*) [38]. Μέχρι και τον Μάιο του 2007, ψάχνανε ακόμη για δύο από αυτούς τους developers.

Η ανάπτυξη του αρχικού κώδικα του Netscape Communicator ήταν, χωρίς αμφιβολία, πιο περίπλοκη από ότι ανέμεναν αρχικά. Η αρχική εκκίνηση ήταν, ήδη, αρκετά άσχημη, διότι αυτό που είχε αποδοθεί ήταν, σε ορισμένες περιστάσεις, ατελές (είχαν απομακρυνθεί όλα τα modules τρίτων κατασκευαστών, για την κυκλοφορία των οποίων δεν είχε δοθεί συναίνεση) και μετά βίας λειτουργούσε. Σαν να μην έφτανε αυτό, εκτός από τα τεχνικά προβλήματα για να μπορεί να λειτουργήσει το Mozilla σε πολλά λειτουργικά συστήματα και πλατφόρμες, υπήρχαν και οι ατέλειες της Netscape Inc., με κύκλους ανάπτυξης (release cycles) που ήταν πολύ μακροσκελείς και αναποτελεσματικοί για τον κόσμο του Διαδικτύου και που δεν έκαναν διάκριση ανάμεσα στα δικά της συμφέροντα και σε εκείνα της κοινότητας που είχε δημιουργηθεί γύρω από το Mozilla. Όλα αυτά ξεκαθάρισαν, ακριβώς έναν χρόνο αργότερα, όταν ένας από τους πλέον ενεργούς προγραμματιστές, τόσο από πριν όσο και μετά το release, ο Jamie Zawinsky, αποφάσισε να πετάξει το γάντι με ένα πικρόχολο γράμμα (*"Resignation and post-mortem"* / *"Παραίτηση και μετά θάνατον νεκροψία"*, 1999) [237] στο οποίο καθιστούσε σαφή την απόγνωση του και τη λύπη του.

Στις 15 Ιουλίου του 2003, η Netscape Inc. (που τώρα ανήκει στην America On Line) ανακοίνωσε ότι δεν επρόκειτο να αναπτύξει τον περιηγητή Netscape και, συνεπώς, δεν θα ελάμβανε πλέον μέρος με ενεργό τρόπο στο Mozilla project.

Υπό μορφή μίας "δήλωσης πλεονασμού" η Netscape ενέκρινε την δημιουργία του Ιδρύματος Mozilla Foundation, το οποίο και υποστήριζε με μία συνεισφορά δύο εκατομμυρίων (2.000.000) δολαρίων. Παρομοίως, όλος ο κώδικας που ήταν υπό την NPL (Netscape's public license) δωρήθηκε στο Ίδρυμα και αναδιανεμήθηκε με τις άδειες που είχαν δημοσιευθεί στο παρελθόν από το Mozilla project: την MPL, την LGPL και την GPL.

Στις 10 Μαρτίου του 2005, το Mozilla Foundation ανακοίνωσε ότι δεν θα δημοσίευε άλλες επίσημες εκδόσεις της Σουίτας εφαρμογών Mozilla Application Suite, την οποία και θα αντικαθιστούσε το Mozilla SeaMonkey, που περιελάμβανε έναν web browser, έναν πελάτη e-mail, ένα βιβλίο διευθύνσεων, έναν επεξεργαστή HTML και έναν πελάτη IRC. Από την άλλη πλευρά, το Mozilla project φιλοξενεί διάφορες ανεξάρτητες εφαρμογές, από τις οποίες οι πιο αξιόλογες περιλαμβάνουν τον Mozilla Firefox (web browser), που σίγουρα είναι η πιο διάσημη, το Mozilla Thunderbird (πελάτης για e-mail και για news), το Mozilla Sunbird (ημερολόγιο), το Mozilla Nvu (HTML editor), το Camino (web browser που σχεδιάστηκε για το Mac OS X) και το Bugzilla (web-based εργαλείο εντοπισμού των bugs).

Με την πάροδο του χρόνου, παρά και τις πολλές αμφιβολίες και τις μακρές περιόδους κατά τις οποίες έμοιαζε ότι ήταν προορισμένο να αποτύχει, το project φαίνεται, τώρα, να πηγαίνει καλά. Χάρη στην ευελιξία και τη φορητότητα των εφαρμογών του και, παρ' ότι απαιτούνται πολλοί πόροι για την εκτέλεσή τους, χρησιμοποιούνται σε πολλές περιστάσεις (γενικά, αλλά και ειδικά ο Firefox) σαν το ζεύγος του OpenOffice.org, στην επιφάνεια εργασίας του τελικού χρήστη.

9.6.2. Ακτινογραφία του Mozilla

Οι αριθμοί που θα συζητήσουμε σε αυτό το κεφάλαιο, αντιπροσωπεύουν μία μελέτη του Firefox, την πιο γνωστή από τις εφαρμογές του project. Σύμφωνα με τις εκτιμήσεις του μοντέλου COCOMO, μία εταιρεία που θα επιθυμούσε να φτιάξει ένα λογισμικό αυτής της κλίμακας θα έπρεπε να επενδύσει περίπου 111 εκατομμύρια δολάρια για να το πετύχει. Ο χρόνος που θα χρειαζόταν θα ήταν περί τα επτά (7) χρόνια και ο μέσος αριθμός των προγραμματιστών που θα έπρεπε να εργάζονται full-time για την εταιρεία, θα ήταν περίπου εκατόν είκοσι (120).

Πίνακας 14. Παρούσα κατάσταση του Mozilla Firefox

Ιστότοπος	www.mozilla-europe.org/es/products/firefox/
Έναρξη του project	2002
Άδεια	MPL/LGPL/GPL
Έκδοση	2.0
Γραμμές πηγαίου κώδικα.	2.768.223
Εκτίμηση κόστους	\$ 111.161.078
Εκτίμηση απαιτούμενου χρόνου ως την ολοκλήρωση	6.87 έτη (82.39 μήνες)
Εκτίμηση του μέσου αριθμού developers	120
Εκτιμώμενος αριθμός developers	50 committers
Εργαλεία υποστήριξης της ανάπτυξης	CVS, mailing lists, IRC, Bugzilla.

Η C++ και η C είναι οι γλώσσες που χρησιμοποιούνται περισσότερο, με αυτή την σειρά προτεραιότητας. Η Perl χρησιμοποιείται και αυτό οφείλεται κυρίως στο ότι τα εργαλεία υποστήριξης της ανάπτυξης που δημιούργησε το Mozilla project, σαν το BugZilla ή το Tinderbox, σχεδιάστηκαν σε αυτή τη γλώσσα. Το εκπληκτικό πράγμα είναι ο μεγάλος αριθμός γραμμών κώδικα, σε γλώσσα assembly, μέσα σε μία εφαρμογή για ένα τελικό χρήστη. Μία επισκόπηση του κώδικα στο αποθετήριο, δείχνει πως, πραγματικά, υπάρχουν πολυπληθή αρχεία σε κώδικα γλώσσας assembly.

Πίνακας 15. Προγραμματιστικές γλώσσες που χρησιμοποιούνται στο Mozilla Firefox

Προγραμματιστική γλώσσα	Γραμμές Κώδικα	Ποσοστό
C++	1.777.764	64.22%
C	896.551	32.39%
Assembler	34.831	1.26%
Perl	26.768	0.97%
Shell	16.278	0.59%
C#	6.232	0.23%
Java	5.352	0.19%
Python	3.077	0.11%
Pascal	459	0.02%

9.7. Το OpenOffice.org

Το OpenOffice.org είναι μία από τις κορυφαίες εφαρμογές στη σημερινή σκηνή του Ελεύθερου Λογισμικού. Πρόκειται για μία πολυπλατφορμική σουίτα εφαρμογών γραφείου, που περιλαμβάνει τις εφαρμογές κλειδί για ένα περιβάλλον επιφάνειας εργασίας γραφείου, όπως έναν επεξεργαστή κειμένου (το Writer), ένα λογιστικό φύλλο (το Calc), ένα πρόγραμμα παρουσίασης (το Impress), ένα επεξεργαστή σχεδίων (το Draw), ένα εργαλείο για την δημιουργία και επεξεργασία μαθηματικών εξισώσεων (το Math) και, τέλος, έναν επεξεργαστή γλώσσας HTML (που περιλαμβάνεται στο Writer). Η διεπαφή που παρέχει το OpenOffice.org είναι ομοιογενής και ενστικτώδης, με εμφάνιση και λειτουργίες παρόμοιες με εκείνες άλλων εφαρμογών γραφείου, ειδικά της πιο πολυχρησιμοποιημένης σήμερα, του Microsoft Office.

Γραμμένο σε C++, το OpenOffice.org περιλαμβάνει το API της JAVA και έχει τα δικά του συστατικά για ενσωματωμένα (embedded) συστήματα, που δίνουν τη δυνατότητα εισαγωγής, για παράδειγμα, πινάκων από ένα λογιστικό φύλλο σε έναν επεξεργαστή κειμένων, με έναν πολύ απλό και διαισθητικό τρόπο. Ένα από τα πλεονεκτήματά του είναι ότι μπορεί να χειρισθεί πολλές μορφές αρχείων, περιλαμβανομένων και εκείνων του Microsoft Office. Οι δικές του εγγενείς μορφές αρχείου, σε αντίθεση με εκείνες της σουίτας γραφείου της Microsoft, βασίζονται στο XML, πράγμα που δείχνει την δέσμευσή του στην ευελιξία, την ευκολία μετατροπής και την διαφάνεια. Στις μέρες μας, το OpenOffice.org έχει μεταφρασθεί σε πάνω από 25 γλώσσες και μπορεί να τρέξει σε Solaris (που είναι και το εγγενές/ native σύστημά του), σε GNU/Linux και σε Windows. Αναμένονται και εκδόσεις για FreeBSD, για IRIX και για Mac OS X, στο όχι και τόσο μακρινό μέλλον.

Το OpenOffice.org πήρε το οριστικό του όνομα (OpenOffice, όπως το γνωρίζει ο καθένας, συν το επίθεμα .org) μετά από μία δικαστική υπόθεση, στην οποία κατηγορήθηκε για παραβίαση δικαιωμάτων εμπορικού σήματος (trademark) εκ μέρους μία άλλης εταιρείας.

9.7.1. Η ιστορία του OpenOffice.org

Στα μισά της δεκαετίας του 1980, στην Ομοσπονδιακή Δημοκρατία της Γερμανίας, ιδρύθηκε η εταιρεία StarDivision, με τον βασικό σκοπό να δημιουργήσει μία σουίτα εφαρμογών γραφείου: το StarOffice. Το καλοκαίρι του 1999, η SUN Microsystems αποφάσισε να αγοράσει την εταιρεία StarDivision και να αναλάβει μία σημαντική δέσμευση προς το StarOffice, με την σαφή πρόθεση να διεκδικήσει μέρος του μεριδίου αγοράς που είχε κατακτήσει η Microsoft την εποχή εκείνη. Τον Ιούνιο του 2000, η εταιρεία κυκλοφόρησε την έκδοση 5.2 του StarOffice, που μπορούσε κανείς να κατεβάσει δωρεάν από το Διαδίκτυο.

Εν πάσει περιπτώσει, η επιτυχία του StarOffice ήταν περιορισμένη, αφού η αγορά είχε κυριαρχηθεί στο έπακρο από το πακέτο γραφείου της Microsoft. Η SUN αποφάσισε να αλλάξει την στρατηγική της και, ανάλογα με ότι συνέβη με το Netscape και το Mozilla project, αποφάσισε να εκμεταλλευθεί το Ελεύθερο Λογισμικό για να προσδώσει σημασία στα συστήματά της και να συνεχίσει την υλοποίησή τους. Κατά συνέπεια, οι μελλοντικές εκδόσεις του StarOffice (ένα ιδιοταγές προϊόν της SUN) επρόκειτο να δημιουργηθούν χρησιμοποιώντας το OpenOffice.org (ένα ελεύθερο προϊόν) σαν πηγή, σεβόμενοι τις προγραμματιστικές διεπαφές της εφαρμογής (τα API) και τις μορφές αρχείου και λειτουργώντας σαν μία πρότυπη υλοποίηση.

9.7.2. Η οργάνωση του OpenOffice.org

Το OpenOffice.org σκοπεύει να έχει μία δομή στη λήψη των αποφάσεων, που να επιτρέπει σε όλα

τα μέλη της κοινότητας να αισθάνονται ότι μπορούν να συμμετέχουν. Κατά συνέπεια, επινοήθηκε ένα σύστημα λήψης αποφάσεων που να συγκεντρώνει την μεγαλύτερη δυνατή ομοφωνία. Το έργο OpenOffice.org αποτελείται από μία σειρά υποέργων, που τα αναλαμβάνουν τα διάφορα μέλη του project, οι βοηθοί και ένας μοναδικός επικεφαλής. Τα μέλη ενός project μπορούν, ασφαλώς, να εργασθούν σε περισσότερα του ενός project, όπως το ίδιο μπορεί να κάνει και ο επικεφαλής. Δεν μπορεί, όμως, κανείς να μπαίνει επικεφαλής σε παραπάνω από ένα έργο, σε οποιαδήποτε δεδομένη χρονική στιγμή. Τα διάφορα projects υποδιαιρούνται σε τρεις κατηγορίες:

- Τα projects που έγιναν αποδεκτά. Αυτά μπορούν να είναι τεχνικής ή μη-τεχνικής φύσεως. Οι επικεφαλής του κάθε αποδεκτού project έχει δικαίωμα μίας ψήφου, όταν πρόκειται να ληφθούν συνολικές αποφάσεις.
- Projects σε μητρική γλώσσα (Native-lang). Αυτά είναι όλα τα projects διεθνοποίησης και επιτόπιας προσαρμογής (*internationalisation and localisation*) του OpenOffice.org. Όπως έχουμε αναφέρει, προς το παρόν, υπάρχουν πάνω από εικοσιπέντε (25) ομάδες που δουλεύουν για τη μετάφραση των εφαρμογών του OpenOffice.org στις διάφορες γλώσσες. Ως ομάδα, τα projects σε εγγενή γλώσσα (*native-lang*) διαθέτουν μία μόνον ψήφο στις συνολικές αποφάσεις.
- Projects υπό εκκώλυση. Αυτά είναι τα projects που προωθούνται από την κοινότητα (σε γενικές γραμμές, πρόκειται για πειραματικά ή μικρά projects). Μπορεί να γίνουν αποδεκτά projects μετά από μία περίοδο έξι (6) μηνών. Πράγματι, η κοινότητα του OpenOffice.org μπορεί να εγγυηθεί ότι τα αποδεκτά projects βασίζονται σε πραγματικό ενδιαφέρον, καθώς το ποσοστό θνησιμότητας των νεόκοπων projects στον κόσμο του Ελεύθερου Λογισμικού είναι εξαιρετικά υψηλό. Συνολικά, τα υπό εκκώλυση projects διαθέτουν μία ψήφο όταν λαμβάνονται οι αποφάσεις.

9.7.3. Ακτινογραφία του OpenOffice.org

Η σουίτα γραφείου OpenOffice.org περιλαμβάνει περίπου 4.000.000 (τέσσερα εκατομ.) γραμμών πηγαίου κώδικα, κατανεμημένου σε 45.000 (σαρανταπέντε χιλιάδες) αρχεία.

Το μοντέλο COCOMO υπολογίζει ότι η εργασία που θα χρειαζόταν για να χτισθεί ένας "κλώνος" του OpenOffice.org θα έπρεπε να παρασχεθεί από 180 (εκατόν ογδόντα) προγραμματιστές οι οποίοι να δουλεύουν full-time για περίπου 8 (οκτώ) χρόνια. Σύμφωνα με τις αποτιμήσεις του μοντέλου COCOMO, το κόστος ανάπτυξης θα ήταν περίπου 215 εκατομμύρια δολάρια.

Τα αποτελέσματα για τα οποία συζητήσαμε σε αυτό το κεφάλαιο προέκυψαν από μία μελέτη του πηγαίου κώδικα της σταθερής έκδοσης 2.1 του OpenOffice.org.

Πίνακας 16. Η παρούσα κατάσταση του OpenOffice.org

Ιστότοπος	http://www.openoffice.org
Έναρξη του project	Ιούνιος 2000 (οι πρώτες ελεύθερες εκδόσεις)
Άδεια	LGPL και SISSL
Έκδοση	2.1
Γραμμές πηγαίου κώδικα	5.197.090
Εκτίμηση κόστους	\$ 215.372.314
Εκτίμηση απαιτούμενου χρόνου ως την ολοκλήρωση	8.83 έτη (105.93 μήνες)
Εκτίμηση του μέσου αριθμού developers	180
Εκτιμώμενος αριθμός developers	200 committers
Εργαλεία υποστήριξης της ανάπτυξης	CVS, mailing lists

Σε ότι αφορά τις χρησιμοποιούμενες προγραμματιστικές γλώσσες από το OpenOffice.org, η κυρίαρχη είναι η C++. Είναι ενδιαφέρον να σημειώσουμε πως η απόκτηση της εταιρείας εκ μέρους της Sun οδήγησε στην ενσωμάτωση μεγάλου μέρους κώδικα της Java στη σουίτα γραφείου, που ξεπέρασε ακόμη και την ποσότητα της γλώσσας C.

Πίνακας 17. Προγραμματιστικές γλώσσες που χρησιμοποιούνται στο OpenOffice.org

Προγραμματιστική γλώσσα	Γραμμές Κώδικα	Ποσοστό
C++	4.615.623	88.81%
Java	385.075	7.41%
C	105.691	2.03%
Perl	54.063	1.04%
Shell	12.732	0.24%
Yacc	6.828	0.13%
C#	6.594	0.13%

9.8. To Red Hat Linux

Το Red Hat Linux υπήρξε μία από τις πρώτες εμπορικές διανομές του GNU/Linux. Σήμερα, είναι πιθανώς μία από τις πιο γνωστές, και μετά βεβαιότητας εκείνη που μπορεί να θεωρείται ως η “κανονική” όλων των εμπορικών διανομών. Η δουλειά των distributors έχει κυρίως να κάνει με εργασίες ενσωμάτωσης (integration tasks) και όχι τόσο πολύ με ανάπτυξη λογισμικού. Τόσο η Red Hat, όσο και οι άλλες διανομές, μπορεί να έχουν developers που εργάζονται γι'αυτές, αλλά το έργο τους είναι δευτερεύουσας σημασίας σε σχέση με τους στόχους της διανομής. Γενικά, υποτίθεται ότι ο βασικός στόχος των διανομών είναι απλώς να παραλαμβάνουν τα πηγαία πακέτα (πρόκειται γενικά για τα αρχεία που έβγαλαν στη κυκλοφορία οι ίδιοι οι developers) και να τα ομαδοποιούν (bundle) με τρόπο που να ικανοποιούνται κάποια κριτήρια (τόσο τεχνικής όσο και οργανωτικής

φύσεως). Το προϊόν αυτής της διαδικασίας είναι μία διανομή: μία σειρά κατάλληλα οργανωμένων ομάδων (bundles) που καθιστούν εφικτό για τον χρήστη να τα εγκαθιστά, να τα απεγκαθιστά και να τα ενημερώνει.

Οι διανομές είναι επίσης υπεύθυνες για την ποιότητα του τελικού προϊόντος, πράγμα που αποτελεί μία πολύ σημαντική πλευρά, αν σκεφθούμε ότι πολλές από τις εφαρμογές που περιλαμβάνονται έχουν αναπτυχθεί από εθελοντές στον ελεύθερό τους χρόνο. Κατά συνέπεια, τα θέματα της ασφάλειας και της σταθερότητας είναι βασικής σημασίας για μία διανομή.

9.8.1. Η ιστορία της Red Hat

Η Red Hat Software Inc. ιδρύθηκε από τους Bob Young και Marc Ewing, το 1994. Ο κύριος σκοπός ήταν να μεταγλωττισθεί και να αξιοποιηθεί εμπορικά μία διανομή GNU/Linux που έφερε το όνομα (και ακόμη αποκαλείται) Red Hat Linux [236]. Βασικά, επρόκειτο για μία πακεταρισμένη έκδοση (bundled version) του τότε περιεχομένου του Διαδικτύου, που περιελάμβανε τεκμηρίωση και υποστήριξη. Η έκδοση 1.0 αυτής της διανομής γεννήθηκε το καλοκαίρι του 1995. Λίγους μήνες αργότερα, το φθινόπωρο, η έκδοση 2.0, περιελάμβανε την τεχνολογία RPM (δόθηκε στη κυκλοφορία ο διαχειριστής πακέτων RPM). Ο διαχειριστής πακέτων RPM έγινε μία από τις de facto προδιαγραφές για τα πακέτα των συστημάτων GNU/Linux. Το 1998, η έκδοση 5.2 της Red Hat δόθηκε στο ευρύ κοινό. Για μία πλήρη ιστορία των ονομάτων των διαφόρων εκδόσεων της Red Hat, παρακαλείσθε να διαβάσετε το "*The truth behind Red Hat names*" [201].

Σημείωση

Από την περίοδο της έκδοσης 1.1 του Linux Standard Base (μία προδιαγραφή που σχεδιάστηκε για να πετύχει δυαδική συμβατότητα μεταξύ των διαφόρων διανομών GNU/Linux, πράγμα που επιμελείται η ομάδα Free Standards Group), το RPM επελέχθη ως ο standard διαχειριστής πακέτων. Το Debian project συνεχίζει με τη δική του μορφή πακέτων, όπως πράττουν πολλές άλλες διανομές που εξαρτώνται από το σύστημα διαχείρισης πακέτων του Debian και προσαρμόζονται στη προτυποποιημένη μορφή, με τη χρήση ενός εργαλείου μετατροπής που λέγεται *alien*.

Πριν προκύψει το σύστημα διαχείρισης RPM, σχεδόν όλες οι διανομές GNU/Linux προσέφεραν τη δυνατότητα εγκατάστασης λογισμικού μέσω μίας διαδικασίας βασισμένης σε μενού (*menu-based procedure*), αλλά το να επιφέρει κανείς μετατροπές σε μία υπάρχουσα εγκατάσταση, ιδιαίτερα το να προσθέσει κανείς νέα πακέτα λογισμικού μετά την εγκατάσταση, δεν ήταν εύκολο πράγμα. Το σύστημα RPM έκανε εφικτό αυτό το βήμα πέραν του state-of-the-art, παρέχοντας στους χρήστες τη δυνατότητα να διαχειρίζονται τα δικά τους πακέτα ("*Maximum RPM. Taking the Red Hat package manager to the limit*", 1998) [83], πράγμα που έδωσε τη δυνατότητα να διαγράφουν, να εγκαθιστούν ή να ενημερώνουν οποιοδήποτε πακέτο λογισμικού που υπήρχε στη διανομή, με ένα πολύ ευκολότερο τρόπο. Το σύστημα πακέτων RPM συνεχίζει να αποτελεί το πλέον χρησιμοποιούμενο σύστημα διαχείρισης πακέτων στις διάφορες διανομές GNU/Linux. Οι στατιστικές των Διανομών Linux, "*Facts and figures*", του 2003 [92], ένας ιστότοπος που περιέχει ποιοτική και ποσοτική πληροφορία για έναν μεγάλο αριθμό διανομών, δείχνουν ότι τον Μάιο του 2003, μία μεγάλη πλειοψηφία (65) από το σύνολο των 118 διανομών που χρησιμοποιήθηκαν για τους υπολογισμούς, έκαναν χρήση του RPM (περίπου το 55% του συνόλου). Σε σύγκριση, η μορφή πακέτων του Debian (γνωστή και ως deb) εχρησιμοποιείτο σε 16 διανομές (περίπου το 14% του συνόλου).

Η Red Hat Inc., πάντως, δεν ήταν γνωστή μόνο για τη διανομή λογισμικού που βασιζόταν στο Linux. Τον Αύγουστο του 1999, η Red Hat εισήχθη στο χρηματιστήριο και οι μετοχές της σημείωσαν την όγδοη καλύτερη επίδοση κέρδους κατά την πρώτη ημέρα εισαγωγής, σε όλη την ιστορία της Wall Street. Τέσσερα χρόνια αργότερα, η αξία των μετοχών της Red Hat είχε

συρρικνωθεί στο ένα εκατοστό της μέγιστης τιμής που είχαν αγγίξει πριν την κρίση των εταιρειών dotcom. Εν τούτοις, η επιτυχής έναρξή τους στην αγορά του Χρηματιστηρίου, έβαλε την Red Hat στα πρωτοσέλιδα περιοδικών και εφημερίδων που δεν ήταν εξειδικευμένα άμεσα σε θέματα IT. Σε κάθε περίπτωση, φαίνεται πως η Red Hat κατάφερε να ξεπεράσει τα προβλήματα που άλλες εταιρείες από τον κόσμο των επιχειρήσεων συνάντησαν με το Ελεύθερο Λογισμικό και οι αριθμοί που δημοσίευσε κατά το πρώτο τέταρτο του 2002, ήταν με θετικό πρόσημο για πρώτη φορά στην ιστορία της.

Ένα άλλο από τα πιο σημαντικά ιστορικά γεγονότα που αφορούν την Red Hat υπήρξε η εξαγορά της Cygnus Solutions, το Νοέμβριο του 1999, μίας εταιρείας που ιδρύθηκε μία δεκαετία πριν ακόμη αποδειχθεί πως ήταν εφικτό να κερδίσει κανείς χρήματα με μία ολοκληρωμένη στρατηγική που να βασίζεται στο Ελεύθερο Λογισμικό ("*Future of Cygnus Solutions. An entrepreneur's account*") [216]. Η Cygnus επέλεξε την περίπλοκη αγορά των μεταγλωττιστών για να αφήσει τη σφραγίδα της. Η εμπορική της στρατηγική βασιζόταν στην ανάπτυξη και την προσαρμογή των εργαλείων GNU για την ανάπτυξη λογισμικού (βασικά το GCC και το GDB) κομμένα και ραμμένα για τις ανάγκες του πελάτη.

Τον Σεπτέμβριο του 2003, η Red Hat αποφάσισε να επικεντρώσει το έργο ανάπτυξης στην εμπορική έκδοση της διανομής της και ανέθεσε την έκδοση της κοινότητας στο Fedora Core, ένα project Ανοικτού Κώδικα ανεξάρτητου από την Red Hat.

Τον Ιούνιο του 2006, η Red Hat εξαγόρασε την εταιρεία JBoss, Inc., μετατρέποντάς την έτσι σε μία εταιρεία επικεφαλής της ανάπτυξης του πιο σημαντικού open source application server, J2EE.

9.8.2. Η παρούσα κατάσταση της Red Hat.

Τρεχόντως, τα πιο σημαντικά προϊόντα της Red Hat Inc. είναι το Fedora Core και το Red Hat Network, μία υπηρεσία ενημέρωσης λογισμικού μέσω Διαδικτύου. Αυτού του είδους οι υπηρεσίες έχουν σχεδιασθεί πιο πολύ έχοντας τον τελικό χρήστη κατά νου, και όχι τόσο πολύ για το επιχειρηματικό περιβάλλον, αλλά είναι ένας καλός τρόπος για να διαφημίζει η Red Hat τον εαυτό της και να ενισχύει την στρατηγική του εμπορικού της ονόματος (*brand strategy*).

Η “πραγματική” εμπορική στρατηγική πολιτική της Red Hat βασίζεται στα προϊόντα που σχεδιάζει για τον επιχειρηματικό κόσμο. Αυτού του είδους τα προϊόντα είναι πολύ λιγότερο γνωστά, αλλά αποτελούν το μεγαλύτερο τμήμα του κύκλου εργασιών της Red Hat, κατά πολύ μεγαλύτερο από εκείνον των πιο δημοφιλών προϊόντων της, με την κυριολεκτική έννοια.

Η Red Hat διαθέτει μία διανομή που είναι προσανατολισμένη προς τον επιχειρηματικό κόσμο, εναρμονισμένη γύρω από έναν εξυπηρετητή εφαρμογών με το όνομα Red Hat Enterprise Linux AS. Οι πελάτες που αγοράζουν αυτό το λογισμικό λαμβάνουν επίσης και υποστήριξη. Το ισοδύναμο του Red Hat Network, για εμπορικούς χρήστες είναι το Red Hat Enterprise Network, που περιλαμβάνει τη διαχείριση συστήματος και την επιλογή για να γίνεται ενημέρωση. Από την άλλη πλευρά, η Red Hat προσφέρει υπηρεσίες συμβουλευτικής IT και ένα πρόγραμμα πιστοποίησης παρόμοιο με εκείνο που προσφέρει η Microsoft στον κόσμο των Windows.

9.8.3. Ακτινογραφία της Red Hat

Η Red Hat ξεπέρασε πρόσφατα το ορόσημο των 50.000.000 γραμμών κώδικα, πράγμα που την αναδεικνύει σε μία από τις μεγαλύτερες διανομές software που υπήρξαν ποτέ, προσπερνώντας,

όπως θα δούμε αργότερα σε αυτό το κεφάλαιο, ακόμη και το μέγεθος των ιδιοταγών λειτουργικών συστημάτων. Η έκδοση Red Hat Version 8.1 περιελάμβανε 792 πακέτα, άρα μπορούμε να υπολογίζουμε ότι η πιο πρόσφατη έκδοση θα έχει περισσότερα από 800 πακέτα, αν λάβουμε υπ' όψη ότι ο αριθμός τείνει να αυξηθεί ελαφρά από έκδοση σε έκδοση.

Όπως και στα προηγούμενα παραδείγματά μας, χρησιμοποιήθηκε το μοντέλο COCOMO για να εκτιμηθεί η επένδυση και η συνολική προσπάθεια που θα ήταν απαραίτητες για τη δημιουργία μίας κατηγορίας λογισμικού της ίδιας κλίμακας. Στη περίπτωση της Red Hat, πάντως, λάβαμε υπ' όψη μας το γεγονός πως επρόκειτο για ένα προϊόν που παρήχθη με τη χρήση μίας σειράς ανεξάρτητων εφαρμογών. Χρησιμοποιήθηκε, κατά συνέπεια, μία ανεξάρτητη εκτίμηση του μοντέλου COCOMO για κάθε ένα από τα πακέτα της Red Hat, και στο τέλος προσθέσαμε όλα τα επί μέρους εκτιμώμενα κόστη και τον αριθμό προσωπικού που θα ήταν απαραίτητα. Για να αναλύσουμε τον βέλτιστο απαιτούμενο χρόνο σχεδιασμού (optimum design time) για την Red Hat, επιλέξαμε το μεγαλύτερο πακέτο, καθώς από θεωρητική άποψη, όλα τα πακέτα είναι ανεξάρτητα και συνεπώς θα μπορούσαν να σχεδιασθούν και συγχρόνως. Για το λόγο αυτό, ο βέλτιστος απαιτούμενος χρόνος για την Red Hat είναι παρόμοιος με εκείνον που προέκυψε για άλλα projects που παρουσιάσαμε σε προηγούμενες παραγράφους αυτού του κεφαλαίου.

Σύμφωνα με το μοντέλο COCOMO, θα χρειαζόταν περίπου 7.5 χρόνια και μία ομάδα developers, που να αποτελείται από έναν μέσο όρο 1800 developers, για να μπορέσουν να σχεδιάσουν την διανομή Red Hat Linux 8., ξεκινώντας από το μηδέν. Το κόστος ολοκλήρωσης μίας τέτοιας ανάπτυξης θα ανέρχονταν κατά προσέγγιση στα 1800 εκατομμύρια δολάρια.

Σημείωση

1800 εκατομμύρια είναι το ποσό που το Ισπανικό Υπουργείο Άμυνας αφιέρωσε για την ανανέωση του στόλου ελικοπτέρων στον τελευταίο ισολογισμό. Από αυτό το ποσό, το μισό θα επενδυθεί για την αγορά 24 ελικοπτέρων, άρα θα μπορούσαμε να πούμε ότι η τιμή της Red Hat ισοδυναμεί με εκείνη 48 μαχητικών ελικοπτέρων. Παρομοίως, 1800 εκατομμύρια δολάρια ήταν το συνολικό κέρδος από την ταινία “Τιτανικός”.

Πίνακας 18. Κατάσταση του Red Hat Linux.

Ιστότοπος	http://www.redhat.com
Έναρξη του project	1993
Άδεια	
Έκδοση	9.0
Γραμμές πηγαίου κώδικα.	Πάνω από 50.000.000
Αριθμός πακέτων	792
Εκτίμηση κόστους	\$ 1.800.000.000
Εκτίμηση απαιτούμενου χρόνου ως την ολοκλήρωση	7.35 έτη (88.25 μήνες)
Εκτίμηση του μέσου αριθμού developers	1800
Εκτιμώμενος αριθμός developers	Υπάλληλοι της Red Hat (γενικά μόνο για integration)
Εργαλεία υποστήριξης της ανάπτυξης	CVS, mailing lists

Καθώς υπάρχουν πολλά πακέτα, η ποικιλία των γλωσσών που χρησιμοποιούνται στη Red Hat είναι

μεγαλύτερη από ότι είδαμε στις σημαντικότερες εφαρμογές Ελεύθερου Λογισμικού. Σε γενικές γραμμές, η συμμετοχή της C είναι πολύ σημαντική, με περισσότερες από το 60% των γραμμών κώδικα. Σε δεύτερη θέση, με περισσότερες από 10.000.000 γραμμές κώδικα, έχουμε την C++, που ακολουθείται σε μεγάλη απόσταση από τη Shell. Είναι ενδιαφέρον να παρατηρήσουμε ότι μετά την Perl έχουμε την Lisp (κυρίως χάρη στη χρήση της στο Emacs), τη γλώσσα μηχανής (*assembly language*), της οποίας το ένα τέταρτο αντιστοιχεί στη γλώσσα με την οποία έρχεται το Linux) και μία γλώσσα της οποίας η χρήση είναι σαφώς σε υποχώρηση, την Fortran.

Πίνακας 19. Προγραμματιστικές γλώσσες που χρησιμοποιούνται στο Red Hat.

Προγραμματιστική γλώσσα	Γραμμές κώδικα	Ποσοστό
C	30.993.778	62.13%
C++	10.216.270	20.48%
Shell	3.251.493	6.52%
Perl	1.106.082	2.22%
Lisp	958.037	1.92%
Assembler	641.350	1.29%
Fortran	532.629	1.07%

9.9. To Debian GNU/Linux

Το Debian είναι ένα ελεύθερο λειτουργικό σύστημα που, για τη διανομή του, χρησιμοποιεί τρεχόντως τον πυρήνα Linux (μολονότι αναμένεται να υπάρξουν στο μέλλον διανομές του Debian βασισμένες και σε άλλους πυρήνες, όπως είναι για παράδειγμα ο πυρήνας "HURD"). Αυτή τη στιγμή (στο 2007) είναι διαθέσιμο για πολλές και διάφορες αρχιτεκτονικές, περιλαμβανομένων και των Intel x86, ARM, Motorola, 680x0, PowerPC, Alpha και του SPARC. Το Debian δεν είναι μόνο η μεγαλύτερη υπάρχουσα διανομή GNU/Linux, αλλά και μία από τις πιο σταθερές και έχει τιμηθεί με πολλά βραβεία, λόγω του ότι προτιμάται από τους χρήστες. Αν και είναι δύσκολο να υπολογίσει κανείς την ακριβή βάση χρηστών του, καθώς το Debian project δεν βγάζει προς πώληση CD ή οποιαδήποτε άλλα media μαζί με το λογισμικό της, πέραν του ότι το λογισμικό της μπορεί να το αναδιανείμει όποιος επιθυμεί, και θα υποθέταμε, με αρκετά εύλογη βεβαιότητα, ότι πρόκειται για μία σημαντική διανομή μέσα στην αγορά του GNU/Linux.

Υπάρχει μία κατηγοριοποίηση στο Debian που εξαρτάται από το είδος της Αδείας και τις απαιτήσεις των πακέτων της διανομής. Ο πυρήνας της διανομής Debian (που είναι το τμήμα που αποκαλείται "κύριο" και που καλύπτει μία μεγάλη ποικιλία πακέτων) συνίσταται αποκλειστικά και μόνο από Ελεύθερο Λογισμικό, σύμφωνα με τις DFSG (*Κατευθυντήριες Γραμμές Ελεύθερου Λογισμικού Debian/ Debian Free Software Guidelines*) [104]. Μπορεί κανείς να την κατεβάσει από το Διαδίκτυο και πολλοί αναδιανομείς την πωλούν σε CD ή και άλλα media.

Οι διανομές Debian δημιουργούνται από σχεδόν 1000 εθελοντές (που είναι γενικά επαγγελματίες IT και ειδικοί). Το έργο αυτών των εθελοντών συνίσταται στο να λαμβάνουν τα πηγαία προγράμματα, στις περισσότερες περιπτώσεις απ' ευθείας από τους αρχικούς δημιουργούς, να τα ρυθμίζουν, να τα μεταγλωττίζουν και να τα ομαδοποιούν (πακετάρουν), κατά τρόπο ώστε ο μέσος χρήστης της διανομής Debian να έχει μόνο να επιλέξει το πακέτο που επιθυμεί και, από εκεί και πέρα, το σύστημα το ίδιο θα το εγκαθιστά χωρίς περαιτέρω προβλήματα. Αυτό που μπορεί, στην

αρχή, να μοιάζει απλό, μπορεί, στη συνέχεια, να αποδειχθεί σύνθετο, μόλις ληφθούν υπ' όψη και άλλοι παράγοντες, όπως οι εξαρτήσεις ανάμεσα στα διάφορα πακέτα (το πακέτο A χρειάζεται το πακέτο B για να μπορεί να λειτουργεί) και οι διάφορες εκδόσεις όλων αυτών των πακέτων.

Το έργο που επιτελούν τα μέλη της κοινότητας του Debian project είναι το ίδιο σαν και εκείνο που φέρουν εις πέρας σε οποιαδήποτε άλλη διανομή: η εναρμόνιση όλων των επί μέρους λογισμικών, εις τρόπον ώστε συνολικά να λειτουργεί όπως πρέπει. Εκτός από το έργο προσαρμογής και πακετοποίησης, οι developers του Debian είναι υπεύθυνοι και για τη συντήρηση της υποδομής που απαιτείται για τις υπηρεσίες που βασίζονται στο Διαδίκτυο (ιστότοπος, online αρχεία, σύστημα διαχείρισης των bug, ταχυδρομικές λίστες βοήθειας, υποστήριξης και ανάπτυξης, κλπ.), για τα διάφορα projects μετάφρασης και διεθνοποίησης, την ανάπτυξη διάφορων εργαλείων ειδικά για το Debian και, γενικά, υπεύθυνοι για ο,τιδήποτε χρειάζεται για να μπορεί η διανομή Debian να λειτουργεί άψογα.

Εκτός από την εθελοντική του φύση, το Debian project διαθέτει και ένα χαρακτηριστικό με μοναδική ιδιαιτερότητα:

το κοινωνικό συμβόλαιο του Debian (http://www.debian.org/social_contract.html) [106]. Αυτό το κείμενο δεν περιγράφει μόνο τους βασικούς σκοπούς του Debian project, αλλά και τα μέσα που θα χρησιμοποιηθούν για την επίτευξή τους.

Το Debian είναι, επίσης, γνωστό για το ότι διαθέτει πολύ αυστηρή πολιτική ως προς τα πακέτα και τις εκδόσεις του, σχεδιασμένη να πετύχει τη βέλτιστη ποιότητα του προϊόντος (είναι το *"Εγχειρίδιο πολιτικής του Debian"*) [105]. Κατ' αυτόν τον τρόπο, υπάρχουν τρεις διαφορετικοί τύποι Debian ανά πάσα στιγμή: μία σταθερή έκδοση, μία ασταθής έκδοση και μία δοκιμαστική έκδοση. Όπως εισηγείται και το ίδιο το όνομα, η σταθερή έκδοση είναι εκείνη που συνιστάται για συστήματα και χρήστες που απαιτούν την απόλυτη σταθερότητα. Το λογισμικό θα πρέπει να υποβάλλεται σε μία περίοδο "παγώματος", κατά τη διάρκεια της οποίας διορθώνονται τα οποιαδήποτε bugs. Ο γενικός κανόνας είναι ότι η σταθερή διανομή Debian δεν θα πρέπει να περιέχει κανένα γνωστό κρίσιμο bug. Από την άλλη πλευρά, η σταθερή αυτή έκδοση δεν περιλαμβάνει συνήθως τις πιο πρόσφατες εκδόσεις του λογισμικού (δηλ. τις πιο πρόσφατες προσθήκες).

Υπάρχουν ακόμη δύο εκδόσεις του Debian που κυκλοφορούν, παράλληλα με τη σταθερή, και προορίζονται για εκείνους που επιθυμούν να έχουν το πιο πρόσφατο λογισμικό. Η ασταθής έκδοση περιλαμβάνει πακέτα που είναι υπό σταθεροποίηση, ενώ η δοκιμαστική έκδοση (test version), όπως δείχνει και το όνομα, είναι εκείνη που έχει και την μεγαλύτερη τάση για προβλήματα και που περιέχει ό,τι πιο καινούργιο από άποψη των τελευταίων λογισμικών. Όταν διεξήχθη η πρώτη μελέτη, η σταθερή έκδοση του Debian ήταν το Debian 3.0 (που ήταν γνωστό και ως Woody), η ασταθής ήταν γνωστή με το υποκοριστικό Sid και η δοκιμαστική ήταν το Sarge. Ο Woody, πάντως, πέρασε από ένα ασταθές στάδιο και, πριν από αυτό, ένα δοκιμαστικό στάδιο. Αυτό είναι σημαντικό, διότι αυτά που θα εξετάσουμε σε αυτό το κεφάλαιο θα περιλαμβάνουν τις διάφορες σθερές εκδόσεις του Debian, από την εποχή κυκλοφορίας της έκδοσης 2.0, το 1998. Έχουμε, για παράδειγμα, το Debian 2.0 (ή Hamm), το Debian 2.1 (ή Slink), το Debian 2.2 (ή Potato) και τέλος, το Debian 3.0 (ή Woody).

Σημείωση

Τα υποκοριστικά των εκδόσεων του Debian αντιστοιχούν στους βασικούς χαρακτήρες της ταινίας κινουμένων σχεδίων Toy story, μία παράδοση που άρχισε, στα μισά μεταξύ αστείου και σοβαρού, όταν η κυκλοφόρησε η έκδοση 2.0 και ο Bruce Perens, ο τότε ηγέτης του project και μετέπειτα ιδρυτής του Open Source Initiative και της έκφρασης Ανοικτός Κώδικας, εργαζόταν για την εταιρεία που σχεδίαζε την ταινία. Για παραπάνω λεπτομέρειες σχετικά με την ιστορία του Debian και γενικότερα για την διανομή Debian, συστήνουμε το "A brief history of Debian" [122].

9.9.1. Ακτινογραφία του Debian

Το Debian GNU/Linux είναι πιθανώς η μεγαλύτερη συλλογή (compilation) Ελεύθερου Λογισμικού που λειτουργεί με έναν συντονισμένο τρόπο και, αναμφισβήτητα, ένα από τα μεγαλύτερα προϊόντα λογισμικού που φτιάχτηκαν ποτέ. Η Έκδοση 4.0, που κυκλοφόρησε τον Απρίλιο του 2007 (με το όνομα Etch), αποτελείται από περισσότερα των 10000 πηγαίων πακέτων, με περισσότερες από 288 εκατομμύρια γραμμές κώδικα.

Ο αριθμός των γραμμών κώδικα στο Debian 3.0 είναι 105 εκατομμύρια. Σύμφωνα με το μοντέλο COCOMO, για να παραχθεί ένα λογισμικό παρόμοιο με αυτά που είναι πακεταρισμένα σε αυτή τη διανομή, θα έπρεπε να δαπανηθεί ένα ποσό περίπου της τάξης των 3.600 εκατομμυρίων δολαρίων. Όπως και με την Red Hat, η προσπάθεια που θα έπρεπε να επενδυθεί για τη δημιουργία του κάθε πακέτου χωριστά έχει υπολογισθεί και οι επί μέρους αριθμοί που προέκυψαν προσετέθησαν όλοι μαζί. Για τον ίδιο λόγο, ο χρόνος που θα χρειαζόταν για την ανάπτυξη του Debian θα ήταν μόνο επτά (7) χρόνια, καθώς όλα τα διάφορα πακέτα θα μπορούσαν να φτιαχτούν ταυτόχρονα. Σε κάθε περίπτωση, ένας μέσος όρος περίπου 4000 developers θα έπρεπε να κινητοποιηθούν κατά τη διάρκεια αυτών των επτά ετών.

Σημείωση

3.600.000 εκατομμύρια δολάρια είναι ο προβλεπόμενος προϋπολογισμός του 6^{ου} Κοινοτικού Πλαισίου Στήριξης για έρευνα και ανάπτυξη της Κοινωνίας της Πληροφορίας. Είναι, επίσης, ίσο με το ποσό που η Ισπανική εταιρεία τηλεπικοινωνιών Telefónica προτίθεται να επενδύσει στη Γερμανία για να προσφέρει υπηρεσίες UMTS.

Πίνακας 20. Κατάσταση του Debian

Ιστότοπος	http://www.debian.org
Έναρξη του project	16/08/1993
Άδεια	Εκείνες που πληρούν τις DFSG
Έκδοση που χρησιμοποιείται	Debian 4.0 (ή Etch)
Γραμμές πηγαίου κώδικα	288.500.000
Αριθμός πακέτων	10.106
Εκτίμηση κόστους	\$ 10.140 εκατομμύρια
Εκτίμηση απαιτούμενου χρόνου ως την ολοκλήρωση	8.84 χρόνια
Εκτιμώμενος αριθμός maintainers	Περίπου 1,500
Εργαλεία υποστήριξης της ανάπτυξης	Mailing lists, Σύστημα αναφοράς bug

Η πιο χρησιμοποιούμενη γλώσσα στο Debian 4.0 είναι η C, με πάνω από το 51% των γραμμών κώδικα. Όπως όμως θα δείξουμε λίγο αργότερα σε αυτό το κεφάλαιο, η σημασία της C μειώνεται με το χρόνο, καθώς το 80% του κώδικα των πρώτων εκδόσεων του Debian, ήταν σε C. Η δεύτερη πιο συχνά χρησιμοποιούμενη γλώσσα, η C++, μοιράζεται ένα καλό κομμάτι της ευθύνης για την παρακμή της C. Πάντως, η C επηρεάστηκε ιδιαίτερα από την άνοδο των γλωσσών scripting σαν την Perl, την Python και την PHP. Από την άλλη μεριά, οι γλώσσες σαν την Lisp ή την Java (που υποαντιπροσωπεύεται στο Debian λόγω της άρνησής του να δεχθεί κώδικα που εξαρτάται από την ιδιοταγή εικονική μηχανή της Sun) καταφέρνουν μερικές φορές να διεισδύσουν.

Πίνακας 21. Προγραμματιστικές γλώσσες που χρησιμοποιούνται στο Debian GNU/Linux 4.0

Προγραμματιστική γλώσσα	Γραμμές Κώδικα (σε εκατομ.)	Ποσοστό
C	155	51%
C++	55	19%
Shell	30	10%
Perl	8.1	2.9%
Lisp	7.7	2.7%
Python	7.2	2.5%
Java	6.9	2.4%
PHP	3.5	1.24%

Ο Πίνακας 22 δείχνει πως αναπτύχθηκαν οι πιο σημαντικές γλώσσες εντός του Debian.

Πίνακας 22. Οι γλώσσες που χρησιμοποιούνται περισσότερο στο Debian

Γλώσσα	Debian 2.0	%	Debian 2.1	%	Debian 2.2	%	Debian 3.0	%
C	19.400.000	76.67	27,800,00	74.89	40,900,000	69.12	66,500,000	63.08
C++	1.600.00	6.16	2.800,000	7.57	5.980.000	10.11	13.000.000	12.39
Shell	645.000	2.55	1.150.000	3.10	2.710.000	4.59	8.635.000	8.19
Lisp	1.425.000	5.64	1.890.000	5.10	3.200.000	5.41	4.090.000	3.8.
Perl	425.000	1.68	774.000	2.09	1.395.000	2.36	3.199.000	3.03
Fortran	494.000	1.96	735.000	1.98	1.182.000	1.99	1.939.000	1.84
Python	122.000	0.48	211.000	0.57	349.000	0.59	1.459.000	1.38%
Tcl	311.000	1.23	458.000	1.24	557.000	0.94	1.081.000	1.02

Υπάρχουν και γλώσσες που ενώ θα μπορούσαμε να θεωρήσουμε ότι ανήκουν στη μειονότητα, εν τούτοις, κατατάσσονται στις ψηλές θέσεις της κατάταξης. Αυτό οφείλεται στο ότι, παρόλο που απαντώνται μόνο σε ένα μικρό αριθμό πακέτων, τα εν λόγω πακέτα, είναι αρκετά μεγάλα. Αυτό ισχύει, ας πούμε, για την Ada, που ενώ περιέχει μόνο τρία πακέτα (το GNAT, έναν μεταγλωττιστή Ada/ το libgtkada, έναν δεσμό με τη βιβλιοθήκη GTK και το ASIS, ένα σύστημα για τη διαχείριση των πηγών Ada), εν τούτοις καλύπτει παραπάνω από 430.000 επί συνόλου 576.000 γραμμών πηγαίου κώδικα που καταμετρήθηκαν στο Debian 3.0 για την Ada. Μία άλλη ανάλογη περίπτωση, είναι εκείνη της Lisp, η οποία εμφανίζεται μόνο στο GNU Emacs και στο XEmacs, αλλά έχει παραπάνω από 1.200.000 γραμμές από τις περίπου 4.000.000 στο σύνολο της διανομής.

9.9.2. Σύγκριση με άλλα Λειτουργικά Συστήματα

Υπάρχει μία παροιμία που λέει ότι οι συγκρίσεις είναι απεχθείς. Αυτό αληθεύει ακόμη παραπάνω στη περίπτωση σύγκρισης Ελεύθερου Λογισμικού με το Ιδιοταγές λογισμικό. Οι αναλυτικές ακτινογραφικές εικόνες που εξετέθησαν για το Red Hat Linux και το Debian, ήταν εφικτές διότι αποτελούν παραδείγματα Ελεύθερου Λογισμικού. Το να έχουμε τη δυνατότητα πρόσβασης στον

πηγαίο κώδικα (αλλά και σε άλλες πληροφορίες που δόθηκαν σε αυτό το κεφάλαιο) είναι βασικής σημασίας για τη μελέτη του αριθμού γραμμών κώδικα των διαφόρων εκδόσεων, πακέτων, προγραμματιστικών γλωσσών, κλπ. Τα πλεονεκτήματα όμως του Ελεύθερου Λογισμικού πηγαίνουν πολύ πέραν αυτών, διότι, επιπρόσθετα, καθιστούν ευκολότερη την ανάλυση εκ μέρους τρίτων ενδιαφερομένων, είτε πρόκειται για ερευνητικές ομάδες είτε για μεμονωμένα άτομα.

Στα ιδιοταγή συστήματα, σε γενικές γραμμές, η διεξαγωγή μίας παρόμοιας μελέτης θα ήταν πλήρως αδύνατη. Πράγματι, οι αριθμοί που παρουσιάζονται εδώ, προέκυψαν από τις εταιρείες που βρίσκονται πίσω από την ανάπτυξη διαφόρων ιδιοταγών λογισμικών, πράγμα που σημαίνει πως δεν είμαστε σε θέση να εγγυηθούμε για την αξιοπιστία τους. Και για να το συμπληρώσουμε αυτό, σε πολλές περιπτώσεις δεν γνωρίζουμε αν αναφέρονται στις πραγματικές γραμμές του φυσικού πηγαίου κώδικα, όπως πράξαμε σε όλο αυτό το κεφάλαιο, ή κατά πόσο περιλαμβάνουν επίσης τις κενές γραμμές και τα σχόλια. Περαιτέρω, δεν γνωρίζουμε μετά βεβαιότητας τι ακριβώς ενσωματώνουν στο λογισμικό τους, το οποίο σημαίνει ότι δεν είμαστε σε θέση να γνωρίζουμε αν κάποιες εκδόσεις των Microsoft Windows περιλαμβάνουν την σουίτα Microsoft Office ή όχι.

Εν πάσει περιπτώσει, λαμβάνοντας υπ' όψη όλα εκείνα που συζητήσαμε σχετικά με αυτά τα θέματα στις προηγούμενες παραγράφους, πιστεύουμε πως έχει ενδιαφέρον να συμπεριλάβουμε αυτή τη σύγκριση, καθώς μας βοηθά να δούμε τη θέση στην οποία βρίσκονται οι διάφορες διανομές Red Hat και Debian, μέσα σε ένα ευρύτερο πλαίσιο. Εκείνο που είναι αδιαμφισβήτητο είναι ότι τόσο το Debian, όσο και η Red Hat, αλλά ιδιαιτέρως η τελευταία, αποτελούν τις μεγαλύτερες συλλογές λογισμικού που γνώρισε ποτέ η ανθρωπότητα ως σήμερα.

Οι αριθμοί που παρατίθενται πιο κάτω, προέρχονται από τον Mark Lucovsky [168] για τα Windows 2000, από την SUN Microsystems [171] για το StarOffice 5.2, και τον Gary McGraw [169] για τα Windows XP και τον Bruce Schneier [200] για όλα τα υπόλοιπα συστήματα. Ο Πίνακας 23 παρέχει μία σύγκριση, πηγαίνοντας από το μικρότερο προς το μεγαλύτερο.

Πίνακας 23. Σύγκριση με ιδιοταγή συστήματα

<i>Σύστημα</i>	<i>Ημερομηνία κυκλοφορίας</i>	<i>Γραμμές Κώδικα (περίπου)</i>
Microsoft Windows 3.1	Απρίλιος 1992	3.000.000
SUN Solaris 7	Οκτώβριος 1998	7.500.000
SUN StarOffice 5.2	Ιούνιος 2000	7.600.000
Microsoft Windows 95	Αύγουστος 1995	15.000.000
Red Hat Linux 6.2	Μάρτιος 2000	18.000.000
Debian 2.0	Ιούλιος 1998	25.000.000
Microsoft Windows 2000	Φεβρουάριος 2000	29.000.000
Red Hat Linux 7.1	Απρίλιος 2001	32.000.000
Debian 2.1	Μάρτιος 1999	37.000.000
Windows NT 4.0	Ιούλιος 1996	40.000.000
Red Hat Linux 8.0	Σεπτέμβριος 2002	50.000.000
Debian 2.2	Αύγουστος 2000	55.000.000
Debian 3.0	Ιούλιος 2002	105.000.000

9.10. To Eclipse

Η πλατφόρμα Eclipse αποτελείται από ένα επεκτάσιμο και ανοικτό *Ολοκληρωμένο Περιβάλλον ανάπτυξης (IDE, Integrated Development Environment)*. Ένα IDE είναι ένα πρόγραμμα που αποτελείται από μία ομάδα εργαλείων που είναι χρήσιμα σε έναν developer λογισμικού. Τα βασικά στοιχεία ενός IDE περιλαμβάνουν: έναν επεξεργαστή (*editor*) κώδικα, ένα μεταγλωττιστή/διερμηνευτή (*compiler/interpreter*) και έναν αποσφαλματωτή (*debugger*). Το Eclipse είναι ένα IDE σε Java και παρέχει πολυάριθμα εργαλεία ανάπτυξης λογισμικού. Υποστηρίζει, επίσης, άλλες γλώσσες προγραμματισμού, σαν την C/C++, την Cobol, την Fortran, την PHP ή την Python. Υπάρχει και η δυνατότητα προσθήκης *plug-ins* στη βασική πλατφόρμα του Eclipse, για να αυξηθεί η λειτουργικότητά του.

Ο όρος Eclipse αναφέρεται στην κοινότητα Ελεύθερου Λογισμικού που αναπτύσσει την πλατφόρμα Eclipse. Αυτό το έργο υποδιαιρείται σε διάφορα επί μέρους projects, που έχουν όλα ως κύριο στόχο την προσφορά μίας στιβαρής, προσαρμόσιμης (*scalable*) και ποιοτικής πλατφόρμας για την ανάπτυξη λογισμικού με το Eclipse IDE. Το όλο έργο συντονίζεται από το Ίδρυμα Eclipse Foundation, που είναι μία Μη Κερδοσκοπική Οργάνωση που δημιουργήθηκε για να προωθήσει και να αναπτύξει την πλατφόρμα Eclipse και να υποστηρίξει τόσο την κοινότητα όσο και το όλο οικοσύστημα του Eclipse.

9.10.1. Η ιστορία του Eclipse

Μεγάλο μέρος της προγραμματιστικής δουλειάς του Eclipse διεξήχθη από την IBM, πριν ακόμη τη δημιουργία του ίδιου του Eclipse project. Ο προκάτοχος του Eclipse ήταν το VisualAge και είχε φτιαχτεί με τη χρήση της γλώσσας Smalltalk, σε ένα περιβάλλον ανάπτυξης ονόματι Envy. Μετά την εμφάνιση της Java κατά τη δεκαετία του 1990, η IBM ανέπτυξε μία εικονική μηχανή που δούλευε τόσο με τη Smalltalk όσο και με τη Java. Η ταχεία ανάπτυξη της Java και τα πλεονεκτήματά της με την εστίαση σε ένα Διαδίκτυο που βρισκόταν σε φάση εκρηκτικής διάδοσης, ανάγκασαν την IBM να εγκαταλείψει τη δική της εικονική μηχανή διπλής λειτουργίας και να χτίσει από την αρχή μία νέα πλατφόρμα βασισμένη στη Java. Το τελικό προϊόν ήταν το Eclipse, το οποίο είχε ήδη κοστίσει στην IBM περίπου 40 εκατομμύρια δολάρια το 2001.

Κατά το τέλος του 2001, η IBM, μαζί με την Borland, δημιούργησαν το Μη Κερδοσκοπικό Ίδρυμα Eclipse foundation, κάνοντας έτσι ένα άνοιγμα προς τον κόσμο του Ανοικτού Λογισμικού. Σε αυτή την κοινοπραξία προσχώρησαν σταδιακά και άλλες σημαντικές διεθνείς εταιρείες ανάπτυξης λογισμικού: η Oracle, η Rational Software, η Red Hat, η SuSE, η HP, η Serena, η Ericsson και η Novell, μεταξύ άλλων. Σημειώθηκαν δύο σημαντικές απουσίες: η Microsoft και η Sun Microsystems. Η Microsoft αποκλείστηκε λόγω του μονοπωλίου της στην αγορά και η Sun Microsystems είχε ήδη το δικό της IDE, το NetBeans, που αντιπροσώπευε τον βασικό ανταγωνιστή του Eclipse.

Πράγματι, το όνομα Eclipse επελέχθη με σκοπό να δημιουργηθεί ένα IDE ικανό να προκαλέσει "έκλειψη του Visual Studio" (της Microsoft) και "έκλειψη Ηλίου" (της Sun Microsystems). Η τελευταία σταθερή έκδοση του Eclipse είναι διαθέσιμη για λειτουργικό σύστημα Windows, Linux, Solaris, AIX, HP-UX και για Mac OS X. Όλες οι εκδόσεις του Eclipse χρειάζεται να έχουν εγκατεστημένη στο σύστημα την Java Virtual Machine (JVM), κατά προτίμηση το JRE (*Java Runtime Environment*) ή το JDK (*Java Developer Kit*) της Sun, οι οποίες, με βάση τα ισχύοντα στις αρχές του 2007, δεν είναι ακόμη πλήρως ελεύθερες (παρόλο που η Sun έχει ανακοινώσει την απελευθέρωση του δικού της JVM).

9.10.2. Παρούσα κατάσταση του Eclipse

Όλο το έργο που ετοιμάστηκε για την Κοινοπραξία Eclipse είναι οργανωμένο σε διάφορα projects. Αυτά τα projects υποδιαιρούνται, με τη σειρά τους, σε subprojects και αυτά τα τελευταία σε επί μέρους υποέργα (*components*). Τα projects υψηλού επιπέδου τα διαχειρίζονται οι Επιτροπές του Eclipse Foundation (*PMC, project management committees*). Ο παρακάτω κατάλογος δείχνει τα projects υψηλού επιπέδου:

- Eclipse: Είναι η βασική πλατφόρμα για τα υπόλοιπα υποέργα. Αυτή η πλατφόρμα θα είναι ελεύθερη, στιβαρή, πλήρης και καλής ποιότητας, για την ανάπτυξη των Rich Client Platforms (RCP) και των ολοκληρωμένων εργαλείων (*Integrated Tools, plug-ins*). Ο πυρήνας runtime της πλατφόρμας Eclipse ονομάζεται Equinox και αποτελεί μία υλοποίηση των προδιαγραφών της OS-GI (Open Services Gateway Initiative), που περιγράφει μία αρχιτεκτονική προσανατολισμένη προς τις υπηρεσίες (*services oriented architecture, SOA*) για τις εφαρμογές.
- Εργαλεία (ETP, Eclipse tools project): Διάφορα εργαλεία και κοινά συστατικά για την πλατφόρμα Eclipse.
- Web (WTP, web tools project): Εργαλεία για την ανάπτυξη διαδικτυακών εφαρμογών και *services oriented architecture, SOA* της JEE (Java Enterprise Edition).
- Test and performance tools project (TPTP): Εργαλεία για την διεξαγωγή δοκιμών και Μετρητές επιπέδου αποδοτικότητας, για να μπορούν οι developers να επιβλέπουν τις εφαρμογές τους και να τις καθιστούν πιο παραγωγικές.
- Αναφορές Web (*BIRT, business intelligence and reporting tools*): Σύστημα δημιουργίας αναφορών Web.
- Modelling (*EMP, Eclipse modelling project*): Μοντελοποιημένα (*Model-based*) Εργαλεία ανάπτυξης.
- Δεδομένα (*DTP, data tools platform*): Υποστήριξη για τεχνολογίες διαχείρισης δεδομένων.
- Embedded Συσκευές (*DSDP, device software development platform*): Εργαλεία για την ανάπτυξη εφαρμογών που πρόκειται να τρέξουν σε συσκευές με περιορισμένο hardware, με άλλα λόγια, σε ενσωματωμένες συσκευές (*embedded devices*).
- Service oriented architecture (SOA, Αρχιτεκτονική προσανατολισμένη στις υπηρεσίες): Εργαλεία για την ανάπτυξη projects προσανατολισμένων στις υπηρεσίες.
- Eclipse Technology: Έρευνα, διάδοση και ανάπτυξη της πλατφόρμας Eclipse.

Οι αρχές που διέπουν την ανάπτυξη της κοινότητας Eclipse είναι οι εξής:

- Ποιότητα: Το λογισμικό που αναπτύσσεται στο πλαίσιο του Eclipse πρέπει να πληροί τις ποιοτικές προδιαγραφές της Τεχνολογίας Λογισμικού.
- Ανάπτυξη: Η πλατφόρμα Eclipse και όλα τα εργαλεία που βασίζονται σε αυτήν, πρέπει να αναπτύσσονται δυναμικά, σύμφωνα με τις ανάγκες των χρηστών.

- Αξιοκρατία: Όσο περισσότερο συνεισφέρει κάποιος, τόσο μεγαλύτερες ευθύνες αναλαμβάνει.
- Οικοσύστημα Eclipse: Θα υπάρχουν διάφοροι πόροι που η κοινότητα Ανοικτού Κώδικα θα δωρίζει στην Κοινοπραξία Eclipse. Αυτοί οι πόροι θα αξιοποιούνται με τρόπους που να ωφελούν την κοινότητα.

Η διαδικασία ανάπτυξης του Eclipse ακολουθεί κάποιες προδιαγεγραμμένες φάσεις. Κατ' αρχάς, υπάρχει μία φάση που ονομάζεται φάση “πριν-από-την-πρόταση”, κατά την οποία ένα άτομο ή μία εταιρεία δηλώνουν το ενδιαφέρον τους για την καθιέρωση ενός project. Αν η πρόταση γίνει αποδεκτή, τότε λαμβάνεται η απόφαση αν θα καταταχθεί ως ένα project (ή subproject) ανωτέρου επιπέδου. Το επόμενο βήμα είναι η επικύρωση του project, από τη σκοπιά της εφαρμοσιμότητας και της ποιότητας. Μετά από μία φάση επώασης του project, θα ακολουθήσει μία τελική αναθεώρηση. Αν το project περάσει αυτή την αναθεώρηση, θα έχει αποδείξει την αξία του ενώπιον της κοινότητας Eclipse και θα περάσει στη φάση υλοποίησης.

9.10.3. Ακτινογραφία του Eclipse

Το Eclipse διανέμεται υπό τους όρους της Άδειας EPL (Eclipse Public License). Αυτή η Άδεια θεωρείται Ελεύθερη από το FSF και από την OSI. Υπό την Άδεια EPL, είναι δυνατή η χρήση, τροποποίηση, αντιγραφή και αναδιανομή των νέων εκδόσεων του αδειοδοτούμενου προϊόντος. Ο προκάτοχος της EPL ήταν η CPL (Common Public License). Η CPL είχε γραφτεί από την IBM, ενώ η EPL αποτελεί έργο της Κοινοπραξίας Eclipse.

Ο υπολογισμός της προσπάθειας που επενδύεται στο Eclipse δεν είναι εύκολη υπόθεση. Αυτό συμβαίνει διότι ο πηγαίος κώδικας από τον οποίον αποτελείται το οικοσύστημα του Eclipse, βρίσκεται κατανεμημένος σε πολυάριθμα projects και αποθετήρια λογισμικού.

Πιο κάτω είναι τα αποτελέσματα της εφαρμογής του μοντέλου COCOMO στην πλατφόρμα Eclipse, που χρησιμοποιήθηκε ως βάση για όλα τα υπόλοιπα plug-ins.

Πίνακας 24. Ανάλυση του Eclipse

Ιστότοπος	http://www.eclipse.org
Έναρξη του project	2001
Άδεια	Eclipse Public License
Έκδοση υπό ανάλυση	3.2.2
Γραμμές Πηγαίου Κώδικα	2.163.932
Αριθμός αρχείων	15.426
Υπολογισμός κόστους	\$ 85.831.641
Εκτίμηση απαιτούμενου χρόνου ως την ολοκλήρωση	6.22 έτη (74.68 μήνες)
Εκτίμηση του μέσου αριθμού developers	102.10
Εκτιμώμενος αριθμός developers	133 committers
Εργαλεία υποστήριξης της ανάπτυξης	CVS, Mailing lists, Σύστημα εντοπισμού bug (Bugzilla)

Ο ακόλουθος πίνακας δείχνει τις προγραμματιστικές γλώσσες που χρησιμοποιούνται στο Eclipse 3.2.2:

Πίνακας 25. Προγραμματιστικές γλώσσες που χρησιμοποιούνται στο Eclipse

<i>Γλώσσα Προγραμματισμού</i>	<i>Γραμμές Κώδικα</i>	<i>Ποσοστό</i>
Java	2.066.631	95.50%
C	85.829	3.97%
Perl	3.224	0.06%
C++	5.442	0.25%
JSP	3.786	0.17%
Perl	1.325	0.06%
Lex	1.510	0.03%
Shell	849	0.04%
Python	46	0.00%
PHP	24	0.00%

10. Άλλοι Ελεύθεροι πόροι

"Αν πρέπει να φτιάξεις μία μηλόπιτα από το μηδέν, θα πρέπει πρώτα να φτιάξεις το σύμπαν."
Carl Sagan

Μπορούν οι ιδέες που βρίσκονται πίσω από τα Ελεύθερα Προγράμματα να επεκταθούν και σε άλλους πόρους; Θα μπορούσαμε να θεωρήσουμε ότι οι άλλοι πόροι πληροφοριών που μπορούν εύκολα να αντιγραφούν ηλεκτρονικά, είναι όμοιοι με τα προγράμματα και ότι θα μπορούσαν να ισχύσουν και σε αυτούς οι ίδιες ελευθερίες, οι ίδιοι κανόνες, τα ίδια μοντέλα ανάπτυξης και τα επιχειρηματικά μοντέλα. Υπάρχουν, όμως, μερικές διαφορές και οι συνέπειες αυτών των διαφορών εξηγούν γιατί δεν αναπτύχθηκαν το ίδιο δυνατά σαν τα προγράμματα υπολογιστή. Οι κύριες διαφορές είναι ότι, στην περίπτωση των προγραμμάτων, το μόνο που έχει να κάνει κάποιος για να λειτουργούν, είναι να τα αντιγράψει, ενώ όταν αντιγράφουμε άλλου είδους πληροφορίες, θα πρέπει αυτές να περάσουν από μία διαδικασία, κατά το μάλλον ή ήττον, δαπανηρή, πριν αρχίσουν να αποβαίνουν χρήσιμες καθ' οιονδήποτε τρόπο, που μπορεί να σημαίνει κάτι όπως η εκμάθηση ενός κειμένου, μέχρι και την φάση παραγωγής του hardware, που περιγράφεται στην κατάλληλη γλώσσα.

10.1. Οι πιο σημαντικοί Ελεύθεροι Πόροι

Στο κεφάλαιο 3.2.5 συζητήσαμε ήδη για την τεκμηρίωση προγραμμάτων και άλλα τεχνικά έγγραφα,. Εδώ θα δούμε άλλα είδη δημιουργίας, που μπορεί να είναι υπό μορφή κειμένου, αλλά δεν σχετίζονται με το λογισμικό, αλλά με τομείς της επιστήμης, της τεχνολογίας ή της τέχνης.

10.1.1. Επιστημονικές Δημοσιεύσεις

Ο τρόπος με τον οποίο η επιστήμη εξελίσσεται οφείλεται, σε μεγάλο βαθμό, στο ότι οι ερευνητές

δουλειάς τους σε εφημερίδες και περιοδικά που έχουν εμβέλεια στο ευρύ κοινό. Χάρη σε αυτή τη διάδοση, οι ερευνητές αναπτύσσουν ένα βιογραφικό που τους επιτρέπει να προχωρήσουν προς ανώτερες ιεραρχικά θέσεις, με μεγαλύτερες ευθύνες, ενώ προσπορίζονται αμοιβές από ερευνητικές συμβάσεις που τους περιέρχονται ακριβώς χάρη στο αυξανόμενο κύρος τους.

Αυτός ο τρόπος διάδοσης επιστημονικών δημοσιεύσεων αντιπροσωπεύει ένα *επιχειρηματικό μοντέλο* που απεδείχθη πολύ παραγωγικό. Για να μπορεί να δουλέψει αυτό το μοντέλο, η ποιότητα της δουλειάς πρέπει να είναι εγγυημένη και οι δημοσιεύσεις και τα άρθρα πρέπει να έχουν ευρύτατη διάδοση. Το εμπόδιο που δεν επιτρέπει τη διάδοση, είναι η μεγάλη ποσότητα των, γενικά υψηλού κόστους, περιοδικών που κυκλοφορούν, που μπορεί να αγορασθεί μόνο με γενναϊόδωρους προϋπολογισμούς. Η ποιότητα εξασφαλίζεται από το γεγονός ότι οι δημοσιεύσεις επιθεωρούνται από ειδικούς ή από ισάξιους και ομότιμους επιστήμονες (*peers*).

Σχετικά με αυτό, ανεδύθησαν πολυάριθμα περιοδικά, ανάμεσα στα οποία θα μπορούσαμε να αναφέρουμε το παλαιόμαχο First Monday ("First Monday: peer reviewed journal on the Internet") [26] ή το Public Library Of Science project (PLOS <http://www.publiibraryofscience.org> [55]).

Ο Κατάλογος των Περιοδικών Ανοικτής Πρόσβασης ("*Directory of Open Access Journals*" [22]) παραθέτει πολλά περισσότερα. Θα έπρεπε να επιτρέπεται σε άλλα πρόσωπα, πλην των συγγραφέων, να δημοσιεύουν τροποποιήσεις σε αυτού του είδους τα άρθρα; Υπάρχουν αντιρρήσεις που ποικίλουν από την πιθανότητα για κάτω του μετρίου ποιότητα, ή την αοριστία και υπεκφυγή (*equivocation*) απόψεων ή αποτελεσμάτων, μέχρι τον κίνδυνο να προκύψουν άνθρωποι που θα αντιγράφουν (*plagiarism*) εύκολα τα άρθρα και θα αναρριχηθούν στις βαθμίδες χωρίς κόπο, στερώντας από τους πραγματικούς συγγραφείς την αναγνώριση και τα πλεονεκτήματα που κέρδισαν με την αξία τους. Η υποχρέωση, όμως, που έχουν όλοι οι συγγραφείς να αναφέρουν τον αρχικό συγγραφέα και να υποβάλλουν το άρθρο προς κρίση από ισότιμους και ισάξιους συναδέλφους (*peer-review*) πριν δημοσιευθεί σε ένα περιοδικό με κύρος, μπορεί να αντισταθμίσει αυτά τα προβλήματα (βλέπε το κεφάλαιο 10.2.2).

Έχει καθιερωθεί μία αναλογία ανάμεσα στο Ελεύθερο Λογισμικό και την Επιστήμη, καθώς το μοντέλο ανάπτυξης του πρώτου απαιτεί την μεγαλύτερη δυνατή διάδοση, ανασκοπήσεις από ισόβαθμους συναδέλφους (*peer-reviews*), που είναι κατά τεκμήριο ειδικοί και την εκ νέου χρήση των αποτελεσμάτων ("*Free software/free science*", 2001) [154].

10.1.2. *Νόμοι και Πρότυπα (standards)*

Υπάρχουν έγγραφα ρυθμιστικής φύσεως, που ορίζουν πως πρέπει να γίνονται τα πράγματα, για να βελτιώνεται η συνύπαρξη των ανθρώπων ή για να μπορούν τα προγράμματα και οι μηχανές να συνεργάζονται. Αυτά τα έγγραφα πρέπει να διαδίδονται ευρέως, πράγμα που σημαίνει ότι οποιαδήποτε εμπόδια θα είναι αντιπαραγωγικά. Για το λόγο αυτό, είναι ευνόητο ότι υφίστανται ειδική μεταχείριση, όπως καταδεικνύεται και στον Ισπανικό Νόμο* περί Πνευματικών Δικαιωμάτων: "Οι νομικές ή ρυθμιστικές προβλέψεις και τα αντίστοιχα προσχέδια, οι δικαστικές αποφάσεις διαφόρων Σωμάτων δικαιοδοσίας (*jurisdictional bodies*), καθώς και πράξεις, συμφωνίες, νομικές συζητήσεις και επίσημες αποφάσεις Δημοσίων οργανισμών, αλλά και οι επίσημες μεταφράσεις όλων αυτών των κειμένων, δεν μπορούν να αποτελέσουν το αντικείμενο πνευματικής ιδιοκτησίας".

*Ο αντίστοιχος Ελληνικός Νόμος, στο Άρθρο 2, Ν.2121/1993, αναφέρει: "Η προστασία του παρόντος νόμου δεν εκτείνεται σε επίσημα κείμενα με τα οποία εκφράζεται η άσκηση πολιτειακής αρμοδιότητας και ιδίως σε νομοθετικά, διοικητικά, ή δικαστικά κείμενα, καθώς και στις εκφράσεις της λαϊκής παράδοσης, στις ειδήσεις και στα απλά γεγονότα ή στοιχεία."

Το τεχνολογικό ισοδύναμο αυτών των νόμων θα ήταν τα Πρότυπα (*standards*). Στον προγραμματισμό, τα πρωτόκολλα επικοινωνίας, τόσο ανάμεσα σε απομεμακρυσμένα μηχανήματα, όσο και ανάμεσα σε διαφορετικά αρθρώματα (*modules*) του ίδιου μηχανήματος, είναι ιδιαίτερα σημαντικά. Είναι προφανές πως δεν θα πρέπει να περιορίζουμε τη διάδοσή τους, ειδικά αν θέλουμε να ανθίσουν τα Ελεύθερα Προγράμματα, να συνεργάζονται με άλλα προγράμματα, αλλά, παρ' όλα ταύτα, τα ρυθμιστικά σώματα που καθορίζουν αυτά τα θέματα, όπως το ISO¹¹ και η ITU¹², παραδοσιακά, πωλούν τους κανονισμούς τους και τα Πρότυπα, ακόμη και σε ηλεκτρονική μορφή και απαγορεύουν την αναδιανομή τους. Αν και αυτό μπορεί να δικαιολογηθεί, μέχρι ενός βαθμού, με τον ισχυρισμό της κάλυψης μέρους των εξόδων, η ελεύθερη διάδοση των προτύπων υπήρξε πολύ αποδοτικότερη. Έτσι συνέβη και με τη περίπτωση των Κατευθυντήριων Οδηγιών (*guidelines*) της W3C¹³ και, ειδικά σε ότι αφορά τα πρότυπα του Διαδικτύου, τα έγγραφα που ονομάζονται RFCs (*Αίτημα για σχόλια/ Request for Comments*), που υφίστανται από τα αρχικά στάδια, υπό διάφορες ηλεκτρονικές μορφές, που μπορεί να διαβαστούν με οποιονδήποτε επεξεργαστή κειμένου ή φόρμας.

Εν πάσει περιπτώσει, η επιτυχία των πρωτοκόλλων του Διαδικτύου δεν οφείλεται μόνο στη διαθεσιμότητά τους. Άλλοι παράγοντες περιλαμβάνουν το *μοντέλο ανάπτυξης*, που είναι πολύ παρόμοιο με εκείνο του Ελεύθερου Λογισμικού, χάρη στο ότι είναι ανοικτό στη συμμετοχή οποιουδήποτε ενδιαφερομένου και τη χρήση των mailing lists και άλλων παρόμοιων στοιχείων. Αυτή η διαδικασία περιγράφεται στο "The Internet standards process - revision 3" [94] και στο "The Tao of IETF: A Novice's Guide to the Internet Engineering Task Force" [136].

⁽¹¹⁾ International Organisation for Standardisation

⁽¹²⁾ International Telecommunications Union

⁽¹³⁾ World Wide Web Consortium

Θα έπρεπε να επιτρέπεται η τροποποίηση των κειμένων των νόμων και των κανονισμών; Προφανώς όχι, αν αυτό οδηγεί σε σύγχυση. Για παράδειγμα, ένα RFC θα έπρεπε μόνο να μπορεί να τροποποιηθεί, για να επεξηγηθεί ή για να προστεθούν διευκρινιστικά σχόλια, ενώ, σύμφωνα με τις συστάσεις της W3C, ούτε καν αυτό δεν επιτρέπεται χωρίς τη ρητή εξουσιοδότηση (<http://www.w3.org/Consortium/Legal/2002/copyright-documents-20021231>) [65].

Οι Άδειες αυτές καθεαυτές αποτελούν, επίσης, νομικά έγγραφα που δεν γίνεται να τροποποιηθούν. Μήπως θα έπρεπε να είναι δυνατή η κατάρτιση νέων κανονισμών που να προέρχονται από τους ήδη υπάρχοντες, χρησιμοποιώντας τα αρχικά έγγραφα; Αυτό μάλλον θα οδηγούσε στη αβίαστη διάδοση διαφόρων παρόμοιων αλλά ασύμβατων κανονισμών, οι οποίοι θα δημιουργούσαν σύγχυση και θα βοηθούσε τις εταιρείες που ήδη κυριαρχούν στην αγορά, να προωθήσουν τις δικές τους ασύμβατες ποικιλίες, όπως πραγματικά συμβαίνει, ειδικά στο χώρο του Διαδικτύου. Παρ' όλα αυτά, σε ότι αφορά την κρατική νομοθεσία, πολύ συχνά οι νόμοι έχουν αντιγραφεί στην κυριολεξία από αντίστοιχους νόμους άλλων χωρών και προσαρμόστηκαν με μικρές τροποποιήσεις στις τοπικές ιδιαιτερότητες.

Υπάρχει ένα επιχειρηματικό μοντέλο για τους νόμους και τους κανονισμούς; Υπάρχουν πολλοί επαγγελματίες οι οποίοι δουλεύουν πάνω στους νόμους, έχοντας την ευθύνη του σχεδιασμού τους, της ερμηνείας τους, και της εφαρμογής τους (νομοθέτες, δικηγόροι, δικαστές, κλπ.). Υπάρχουν εργαστήρια που εκδίδουν τα πιστοποιητικά συμμόρφωσης για τους κανονισμούς. Τα ρυθμιστικά σώματα υφίστανται, ή θα πρέπει να υφίστανται, στηριζόμενα στις συνδρομές των μελών τους, τα οποία επιθυμούν την προώθηση των προτύπων, διότι, για παράδειγμα, η όλη τους επιχειρηματική

δραστηριότητα βασίζεται σε προϊόντα που λειτουργούν καλά και με άλλα διαφορετικά συστήματα.

Με τον ίδιο τρόπο που είναι βολικό να υπάρχει ένας ορισμός του Ελεύθερου Λογισμικού ή του Ανοικτού Λογισμικού, είναι εξ' ίσου απαραίτητο να υπάρχει και ένας λειτουργικός ορισμός για τα Ανοικτά Πρότυπα. Ο Bruce Perens (<http://perens.org/OpenStandards>) [15] πρότεινε τον ακόλουθο ορισμό, που βασίζεται στις εξής αρχές:

- 1) Διαθεσιμότητα: Αν είναι δυνατόν, τα Ανοικτά Πρότυπα πρέπει να είναι διαθέσιμα για να μπορούν όλοι να τα διαβάσουν και να τα εφαρμόσουν.
- 2) Μεγιστοποίηση της δυνατότητας επιλογής του τελικού χρήστη.
- 3) Τα Ανοικτά Πρότυπα πρέπει να είναι ελεύθερα προς εφαρμογή από όλους, χωρίς καταβολή δικαιωμάτων εκχώρησης ή αμοιβής (*royalty* or *fee*). Μπορεί να προβλέπεται ένα μικρό αντίτιμο για την έκδοση των πιστοποιητικών συμμόρφωσης, αν και ο Bruce Perens συνιστά πως θα έπρεπε να υπάρχουν διαθέσιμα και εργαλεία αυτοπιστοποίησης.
- 4) Καμία διάκριση υπέρ οποιουδήποτε από τους υλοποιητές (*implementer*).
- 5) Επεκτάσεις ή Δευτερεύουσες Άδειες (*subset permissions*), μη πιστοποιήσιμες.
- 6) Αποφυγή επιθετικών πρακτικών εκ μέρους των κυρίαρχων κατασκευαστών. Όλες οι ιδιοταγείς επεκτάσεις πρέπει να διαθέτουν και μία υλοποίηση με Ανοικτά Πρότυπα.

10.1.3. Εγκυκλοπαίδειες

Το 1999, ο Richard Stallman πρότεινε την ιδέα μίας ελεύθερης εγκυκλοπαίδειας (*"The free universal encyclopaedia and learning resource"*, 2001) [210] σαν ένα μηχανισμό για την αποφυγή του σφετερισμού της γνώσης και για την προσφορά γενικευμένης πρόσβασης στη μάθηση και τα σχετικά κείμενα. Θα φτιαχνόταν από άρθρα που θα παρείχε η κοινότητα, χωρίς κεντρικό έλεγχο, όπου διάφοροι άνθρωποι θα ανέλαμβαναν διαφορετικές αρμοδιότητες, περιλαμβανομένης – σαν σύσταση, αλλά όχι σαν υποχρέωση- και εκείνης της επιθεώρησης (*revising*) ή του ελέγχου των άρθρων. Μία τέτοια εγκυκλοπαίδεια δεν θα περιείχε μόνον κείμενα, αλλά και πολυμεσικό περιεχόμενο καθώς και ελεύθερο εκπαιδευτικό λογισμικό.

Αυτό έγινε πραγματικότητα, χάρη στο ότι προέκυψαν διάφορες πρωτοβουλίες. Για παράδειγμα, η Nupedia (<http://www.nupedia.com>) [178] προσπάθησε να φτιάξει μία ποιοτική εγκυκλοπαίδεια, αλλά η προσπάθεια απέτυχε, ίσως διότι απαιτούσε τη χρήση μίας μορφής προτύπων που ήταν σχετικά δύσκολη στην εκμάθηση (TEI), αν και ο κύριος λόγος ήταν μάλλον διότι υπήρχε η απαίτηση υποβολής όλων των άρθρων για επεξεργασία και επιθεώρηση από επιστήμονες, καθώς και για έλεγχο του στυλ, κλπ.

Ο διάδοχος της Nupedia, που σημείωσε πολύ μεγαλύτερη επιτυχία, ήταν η Wikipedia (<http://www.wikipedia.org>) [69]. Η Wikipedia είναι μία ελεύθερη και σε πολλές γλώσσες εγκυκλοπαίδεια, βασισμένη στην τεχνολογία *wiki*. Η Wikipedia γράφεται συνεργατικά, από εθελοντές και η μεγάλη πλειονότητα των άρθρων μπορούν να τροποποιηθούν από οποιονδήποτε που διαθέτει έναν περιηγητή διαδικτύου (*web browser*). Η επιτυχία της βασίζεται στη δομή της, που είναι πιο ευέλικτη από την άποψη της δυνατότητας επεξεργασίας, πράγμα που αποκλείει τα εμπόδια που είχε θέσει η Nupedia και που την φέρνουν πιο κοντά σε εκείνο που είχε στο μυαλό του

ο Stallman. Η λέξη *wiki* προέρχεται από τα Χαβανέζικα “*wiki wiki*” (σημαίνει “γρήγορα”). Η τεχνολογία *Wiki* επιτρέπει σε οποιονδήποτε να επεξεργάζεται ένα κείμενο, χρησιμοποιώντας το σύστημα δομημένου κειμένου (structured text system), που είναι εξαιρετικά απλό, σύμφωνα με όσα είδαμε στο κεφάλαιο 8.6.2. Τον Φεβρουάριο του 2007, ο αριθμός των άρθρων της Wikipedia στην Αγγλική γλώσσα ήταν παραπάνω από 1.500.000.

Σημείωση

Η Wikipedia είναι ένα project μίας μη κερδοσκοπικής οργάνωσης, της Wikimedia, η οποία περιλαμβάνει, επίσης, τα εξής επί μέρους projects, που βασίζονται στο ίδιο μοντέλο της Wikipedia:

- Wiktionary (<http://www.wiktionary.org>) [66]. Αυτό είναι ένα συνεργατικό project που στοχεύει να δημιουργήσει ένα ελεύθερο πολυγλωσσικό λεξικό, με ορισμούς, ετυμολογίες και προφορές, στις αντίστοιχες γλώσσες.
- Wikibooks (<http://www.wikibooks.org/>) [67]. Αυτό είναι ένα project που έχει ως στόχο τη δωρεάν προσφορά βιβλίων, εγχειριδίων, σεμιναρίων ή άλλων παιδαγωγικών κειμένων, προς οποιονδήποτε τα χρειάζεται.
- Wikiquote (<http://www.wikiquote.org>) [70]. Πρόκειται για μία συλλογή διάσημων φράσεων σε όλες τις γλώσσες, που περιλαμβάνει και τις πηγές, όταν αυτές είναι γνωστές.
- Wikisource. Είναι μία βιβλιοθήκη πρωτοτύπων, γνησίων κειμένων, που ανήκουν στη δημόσια σφαίρα (public domain) ή που έχουν εκδοθεί υπό την GFDL (GNU free documentation license).
- Wikispecies (<http://species.wikimedia.org/>) [71]. Πρόκειται για μία ανοικτή συλλογή ζωικών ειδών, φυτικών ειδών, μυκήτων, βακτηριδίων και κάθε γνωστής έμβιας μορφής.
- Wikinews (<http://wikinews.org/>) [68]. Είναι μία πηγή που περιέχει ελεύθερα νέα, στην οποία οι χρήστες είναι και οι εκδότες.
- Commons (<http://commons.wikimedia.org/>) [19]. Πρόκειται για ένα ελεύθερο αποθετήριο εικόνων και πολυμεσικού περιεχομένου.
- Wikiversity (<http://wikiversity.org/>) [72]. Πρόκειται για μία ανοικτή και ελεύθερη εκπαιδευτική πλατφόρμα, που βασίζεται σε διδακτικά projects όλων των εκπαιδευτικών επιπέδων.
- Meta-Wiki (<http://meta.wikimedia.org/>) [48]. Είναι ο ιστότοπος που υποστηρίζει όλα τα projects του Wikimedia Foundation.

Θα πρέπει να αναφέρουμε και την *Concise Encyclopedia of Mathematics*, που όμως έχει μία πιο περιορισμένη αντίληψη του τι σημαίνει “ελεύθερο” (μπορεί να τη συμβουλευθεί κανείς μόνο μέσω Διαδικτύου) και ένα μοντέλο ανάπτυξης όπου είναι απαραίτητη η υποβολή όλων των συνεισφορών προς μία εκδοτική επιτροπή, πριν τη δημοσίευση.

10.1.4. Σειρές μαθημάτων

Με τον ίδιο σκοπό όπως εκείνον των εγκυκλοπαιδειών, είναι δυνατόν να παραχθούν ελεύθερα εκπαιδευτικά υλικά, όπως σημειώσεις, διαφάνειες, ασκήσεις, βιβλία, ή διδακτικό λογισμικό. Υπάρχει μία τάση θεώρησης των Πανεπιστημίων σαν ένα είδος επιχειρήσεων που παράγουν και πωλούν γνώση, πράγμα που έρχεται σε αντίφαση με τις βασικές τους αρχές. Οι λόγοι για τους οποίους ένα Πανεπιστήμιο μπορεί να θέσει αυτό το υλικό στη διάθεση όλων είναι οι εξής:

- Εκπλήρωση της αποστολής του, ως ενός θεσμού που διαδίδει τη γνώση.
- Λόγω του χαμηλού κόστους που απαιτείται για να καταστήσει το υπάρχον υλικό διαθέσιμο παγκοσμίως.
- Αυτό το υλικό δεν μπορεί να υποκαταστήσει τη διδασκαλία πρόσωπο-με -πρόσωπο.

- Η ιδέα ότι αυτό το υλικό μπορεί να λειτουργήσει σαν μία μορφή διαφήμισης που θα προσελκύσει φοιτητές και θα συνεισφέρει στο συνολικό κύρος του Πανεπιστημίου.
- Δυνατότητα δημιουργίας μίας κοινότητας διδασκόντων που θα επιθεωρούν και θα βελτιώνουν αυτό το υλικό.

Η πιο επιφανής προσπάθεια σε αυτό τον τομέα είναι εκείνη του MIT (<http://ocw.mit.edu>) [174], που έχει σκοπό να καταστήσει διαθέσιμους πάνω από 2000 καλά ταξινομημένους πόρους, με τρόπο συνεπή και ομοιογενή.

10.1.5. Συλλογές και Βάσεις δεδομένων

Αυτή καθ' εαυτή η συλλογή πληροφοριών, σύμφωνα με κάποια προκαθορισμένα κριτήρια, το να τα οργανώσει κανείς και να τα καταστήσει διαθέσιμα, αποτελεί ένα προϊόν που συνιστά πολύτιμη πληροφορία, ανεξαρτήτως από τις ίδιες τις πληροφορίες, οι οποίες, λοιπόν, αποτελούν το προϊόν των δημιουργών τους και, κατά συνέπεια, υπόκεινται σε περιορισμούς της ελευθερίας πρόσβασης, τροποποίησης ή και αναδιανομής του περιεχομένου. Άρα, αν επιθυμούμε να έχουμε ελεύθερη πληροφορία, μπορεί επίσης να θέλουμε και ελεύθερες συλλογές.

Για παράδειγμα, μπορεί να θέλουμε να κατατάξουμε κάποιες σημαντικές πληροφορίες στο Διαδίκτυο, οργανώνοντας και σχολιάζοντας τους συνδέσμους (*links*). Αυτό ακριβώς κάνει το ODP (Open Directory Project <http://dmoz.org> [109]). Το λειτουργεί η Netscape και το συντηρούν εθελοντές editors, οργανωμένοι σύμφωνα με μία ιεραρχική δομή. Μπορεί να αντιγραφεί ο πλήρης κατάλογος υπό μορφή RDF και μπορεί και να αναδημοσιευθεί με κάποιες τροποποιήσεις, όπως κάνει η Google, αλλά και πολλές άλλες μηχανές αναζήτησης που το εκμεταλλεύονται. Η Netscape, στην οποία ο κατάλογος ανήκει, εγγυάται ένα κοινωνικό συμβόλαιο "*Open Directory Project social contract*"[53] εμπνευσμένο από εκείνο της Διανομής Debian (http://www.debian.org/social_contract.html) [106], που ενθαρρύνει και διευκολύνει τις εξωτερικές συνεισφορές, διασφαλίζοντας ότι το Open Directory Project θα παραμείνει πάντα ελεύθερο, με δημόσια ανοικτές πολιτικές, αυτοδιοικούμενο από την κοινότητα και τους χρήστες σε πρώτη προτεραιότητα.

Άλλα παραδείγματα συλλογών που μπορεί να μας ενδιαφέρουν είναι οι διανομές Ελεύθερου Λογισμικού, με τα προγράμματα κατάλληλα τροποποιημένα, ούτως ώστε να ταιριάζουν τέλεια μεταξύ τους και, προμεταγλωττισμένα, για να μπορούν να τρέχουν εύκολα.

10.1.6. Το Hardware

Σε ότι αφορά το hardware, υπάρχουν δύο βασικά θέματα που εμπλέκονται με την ελευθερία. Το πρώτο είναι η ανάγκη να είναι ελεύθερες οι διεπαφές (*interfaces*) και οι ομάδες εντολών (*instruction sets*), με τέτοιο τρόπο που να μπορεί οποιοσδήποτε να δημιουργήσει ένα **device handler** ή έναν μεταγλωττιστή για μία αρχιτεκτονική.

Το δεύτερο θέμα είναι ότι θα πρέπει να υπάρχει διαθέσιμη επαρκής πληροφορία και ισχύς για την αναπαραγωγή του σχεδίου του υλικού (*hardware design*), για να μπορεί να τροποποιηθεί και να συνδυασθεί με άλλα. Τα σχέδια αυτά μπορούν να θεωρηθούν σαν ένα software σε μια κατάλληλη γλώσσα (VHDL, Verilog, κλπ). Το να τα κάνεις να δουλεύουν, πάντως, δεν είναι εύκολη υπόθεση, καθώς θα πρέπει να κατασκευασθούν, διαδικασία δαπανηρή και αργή. Εν πάσει περιπτώσει, υπάρχουν κάποιες πρωτοβουλίες προς αυτή την κατεύθυνση, ανάμεσα στις οποίες θα μπορούσαμε να αναφέρουμε τα OpenCores (<http://www.opencores.org>) [52], για τα ολοκληρωμένα κυκλώματα.

10.1.7. Λογοτεχνία και Τέχνη

Για να ολοκληρώσουμε την ανάλυσή μας για τους Ελεύθερους πόρους, δεν μπορούμε να ξεχάσουμε την Τέχνη και την Λογοτεχνία, των οποίων ο απώτατος σκοπός δεν είναι τόσο στη σφαίρα της χρηστικότητας, όσο στη σφαίρα της αισθητικής. Τι λόγους μπορεί να έχει ένας καλλιτέχνης για να δίνει στους ανθρώπους την ελευθερία να αντιγράφουν, να τροποποιούν, ή και να αναδιανέμουν το έργο του; Από τη μία πλευρά, μπορεί να τον βοηθήσει να γίνει γνωστός και να διαδοθεί το έργο του, πράγμα που θα του δώσει τη δυνατότητα απόκτησης εισοδήματος από άλλες δραστηριότητες, όπως συναυλίες ή παραγγελίες, και από την άλλη, μπορεί να προωθήσει τον πειραματισμό και τη δημιουργικότητα. Στη Τέχνη, έχουμε τις ίδιες περιστάσεις όπως στα τεχνολογικά θέματα. Η καινοτομία είναι κάτι το αθροιστικό (*incremental*) και μπορεί ενίοτε να είναι δύσκολο να διακρίνει κανείς ανάμεσα στη λογοκλοπή (*plagiarism*) και σε ένα έργο που είναι αντιπροσωπευτικό ή που ακολουθεί μία καλλιτεχνική κίνηση ή μία τάση.

Η δημιουργία και η ερμηνεία δεν είναι, προφανώς, το ίδιο πράγμα, όπως δεν είναι το ίδιο ούτε η μουσική με τη λογοτεχνία. Η μουσική, η ζωγραφική, η φωτογραφία και ο κινηματογράφος μοιάζουν πολύ με τα προγράμματα, με την έννοια ότι μπορεί να τα κάνεις να “δουλέψουν” κατ’ ευθείαν και άμεσα σε έναν υπολογιστή, ενώ δεν ισχύει το ίδιο με τη γλυπτική, για παράδειγμα.

Δεν υπάρχουν πολλές πρωτοβουλίες ανοικτών πηγών (*open source*) στο χώρο της Τέχνης και της Λογοτεχνίας και, οι λίγες που υπάρχουν, είναι πολύ διαφορετικές μεταξύ τους. Θα μπορούσαμε να αναφέρουμε τη συλλογή μυθιστορημάτων της Wu Ming (<http://www.wumingfoundation.com>) [29].

10.2. Άδειες για άλλους Ελεύθερους Πόρους

Οι Άδειες για το Ελεύθερο Λογισμικό υπήρξαν πηγή εμπνεύσεως και για άλλους πόρους πνευματικής φύσεως, με τέτοιο τρόπο που πολλές από αυτές υιοθετήθηκαν άμεσα, ειδικά σε ότι αφορά τα κείμενα, ενώ σε άλλες περιστάσεις, τροποποιήθηκαν ελαφρώς, όπως συνέβη με την πρωτοποριακή Άδεια Open Audio (http://www.eff.org/IP/Open_licenses/eff_oal.html) [114]. Οι πιο πολλές από αυτές τις Άδειες είναι του τύπου *copyleft*, εφ’ όσον επιτρέπουν την κυκλοφορία παράγωγων έργων.

Η άδεια GNU για ελεύθερα κείμενα (*free documentation*) (βέπε κεφάλαιο 10.2.1) χρησιμοποιήθηκε και χρησιμοποιείται συχνά, για κάθε είδους κείμενα, αν και σταδιακά αυξάνεται η αποδοχή των Αδειών Creative Commons (βλέπε κεφάλαιο 10.2.2).

Πράγματι, οι άδειες λογισμικού (η GPL και η LGPL) έχουν χρησιμοποιηθεί μέχρι και για hardware, αν και αυτό το κομμάτι είναι σύνθετο και δύσκολο να συμφιλιωθεί με την υπάρχουσα νομοθεσία. Τα σχέδια και τα διαγράμματα μπορούν, πράγματι, να χρησιμοποιηθούν χωρίς να αντιγράφονται στην κυριολεξία, για να εξαχθούν ιδέες που θα χρησιμοποιηθούν σε νέα κλειστά σχέδια. Η Άδεια OpenIPCore Hardware General Public License ("*OpenIPCore hardware general public license*") [155], για παράδειγμα, ορίζει ότι δεν επιτρέπεται αυτός ο σφετερισμός, αλλά η νομική εγκυρότητα αυτού του εγγράφου είναι αμφισβητήσιμη [209]. Ο μόνος δυνατός τρόπος για την προστασία αυτών των ιδεών, είναι η χρήση κάποιας μορφής ελεύθερου διπλώματος ευρεσιτεχνίας (*free patent*), κάτι που, όμως, δεν έχει ακόμη αναπτυχθεί και που παραμένει ανέφικτο για εκείνους που δεν προτίθενται ή που δεν μπορούν να ξεκινήσουν μία επιχείρηση βασισμένη σε αυτές τις ιδέες.

10.2.1. Άδεια GNU free documentation license

Μία από τις πιο γνωστές άδειες τύπου *copyleft* για τεχνικά κείμενα, είτε αυτά σχετίζονται με προγράμματα ή με οτιδήποτε άλλο, είναι εκείνη του Ιδρύματος Ελεύθερου Λογισμικού (Free Software Foundation). Όταν ο Richard Stallman συνειδητοποίησε ότι ένα κείμενο δεν είναι το ίδιο πράγμα με ένα πρόγραμμα, τότε προώθησε τη δημιουργία μίας αδειάς για τα κείμενα που συνοδεύουν τα προγράμματα καθώς και για άλλα κείμενα τεχνικής ή διδακτικής φύσεως.

Για να καταστεί πιο εύκολη η ανάπτυξη των παράγωγων εκδόσεων, πέραν των αδιαφανών αντιγράφων (*opaque copies*), πρέπει να είναι διαθέσιμο, σε όποιον το χρειάζεται, και ένα διαφανές αντίγραφο (*transparent copy*) του κειμένου, όπως εξηγήσαμε στο κεφάλαιο 3.2.5, κατ' αναλογία με εκείνο που ισχύει ανάμεσα στον πηγαίο κώδικα και τα αντικείμενα (*objects*) ενός προγράμματος.

Ένας από τους λόγους για να υπάρχει μία άδεια, είναι για να ορισθεί ποιος είναι ο συγγραφέας/δημιουργός, αλλά και για να διασφαλισθεί ότι δεν θα παραποιηθούν οι ιδέες ή οι απόψεις που εκφράζει αυτός ο δημιουργός/ συγγραφέας. Αυτός είναι ο λόγος για τον οποίο τα παράγωγα έργα πρέπει να φέρουν έναν τίτλο στο εξώφυλλο, διαφορετικό από εκείνον των προηγούμενων εκδόσεων (εκτός και αν παραχωρήθηκε ρητή άδεια) και θα πρέπει να αναφέρει ρητά το μέρος όπου μπορεί να προμηθευθεί κανείς το πρωτότυπο. Θα πρέπει να παρατίθενται και τα ονόματα των βασικών συγγραφέων των αρχικών κειμένων, καθώς και τα ονόματα των ανθρώπων που επέφεραν οποιεσδήποτε τροποποιήσεις, και τέλος, πρέπει να διαφυλάσσονται όλες οι σημειώσεις περί δικαιωμάτων πνευματικής ιδιοκτησίας. Παρομοίως, όλες οι αναγνωρίσεις και οι αφιερώσεις πρέπει να διατηρούνται και, αν υπάρχει, και το ιστορικό θα πρέπει να γίνεται σεβαστό όταν προστίθενται νέες εκδόσεις. Είναι δυνατόν (και αυτό είναι το κομμάτι της Αδειάς που συγκέντρωσε την περισσότερη κριτική) ακόμη και να ορισθούν κάποια κεφάλαια ως αμετάβλητα και να καλύψει κείμενα που κανείς να μη μπορεί να τα τροποποιήσει ή να τα διαγράψει, αν και η άδεια επιτρέπει μόνο στα κείμενα *τεχνικής* φύσεως να θεωρηθούν ως αμετάβλητα, και στα οποία η Άδεια αναφέρεται με τον όρο “Δευτερεύοντα Κεφάλαια” (*secondary sections*).

Αυτή η Άδεια προκάλεσε μεγάλη αμφισβήτηση στον κόσμο του Ελεύθερου Λογισμικού, σε σημείο που η διανομή Debian λαμβάνει πλέον υπ' όψη της (κατά τη στιγμή της δημοσίευσης αυτού του βιβλίου) τη δυνατότητα να απομακρύνει από το Debian τα περιεχόμενα που είναι κάτω από αυτή την Άδεια, ή να τα θεωρήσει ως μη ελεύθερα και να τα ταξινομήσει ως μη-επίσημα. Αν και δεν υπάρχουν αμετάβλητα κεφάλαια, αφού τα παράγωγα έργα πρέπει να υπόκεινται στους όρους της ίδιας Αδειάς, είναι σημαντικό να θυμόμαστε ότι θα μπορούσαν να προστεθούν εκ των υστέρων.

Αναφέρεται το επιχείρημα, για παράδειγμα, ότι μπορεί να υπάρχουν εσφαλμένα ή παρωχημένα αμετάβλητα κεφάλαια, τα οποία, παρ' όλα αυτά, πρέπει να διατηρηθούν. Εν πάσει περιπτώσει, η άδεια είναι ασύμβατη με τις Καθοδηγητικές γραμμές του Debian για το Ελεύθερο Λογισμικό (http://www.debian.org/social_contract.html#guidelines) [104], αλλά η ερώτηση περιστρέφεται μάλλον γύρω από το εάν τα κείμενα θα πρέπει να ακολουθούν αυτές τις κατευθυντήριες οδηγίες (για παράδειγμα, δεν μπορούν να τροποποιηθούν ούτε τα κείμενα των αδειών).

Συμβουλή

Οι πρώτες εκδόσεις αυτού του κειμένου είχαν καλυφθεί υπό την Άδεια GFDL, αλλά στη συνέχεια, οι συγγραφείς αποφάσισαν να χρησιμοποιήσουν παράλληλα και την Άδεια Creative Commons (βλέπε κεφάλαιο 10.2.2), που είναι πιο κατάλληλη για τα χαρακτηριστικά ενός βιβλίου. Έτσι, αυτό το κείμενο είναι ένα έργο υπό διπλή αδειοδότηση.







10.2.2. Άδειες Creative Commons

Η Creative Commons (<http://creativecommons.org>) [21] είναι μία Μη-Κερδοσκοπική οργάνωση που ιδρύθηκε το 2001, από ειδικούς στην πνευματική ιδιοκτησία και τους νόμους της Κοινωνίας της Πληροφορίας, με σκοπό τη διευκόλυνση της δημιουργίας, διατήρησης και της προσβασιμότητας των πόρων πνευματικής φύσεως, που παραχωρούνται στην κοινότητα με πολλούς τρόπους. Βασίζεται στην ιδέα ότι μερικοί άνθρωποι μπορεί να μην επιθυμούν να κάνουν χρήση των δικαιωμάτων πνευματικής ιδιοκτησίας που ο νόμος τους παραχωρεί, καθώς κάτι τέτοιο θα εμπόδιζε την περαιτέρω ευρεία διάδοσή τους .

Οι πρώτες Άδειες Creative Commons για δημιουργικά έργα, από τις οποίες υπήρχαν διάφορες εκδόσεις, πρωτοεμφανίσθηκαν γύρω στα τέλη του 2002. Αυτές οι Άδειες είχαν σχεδιασθεί να είναι:

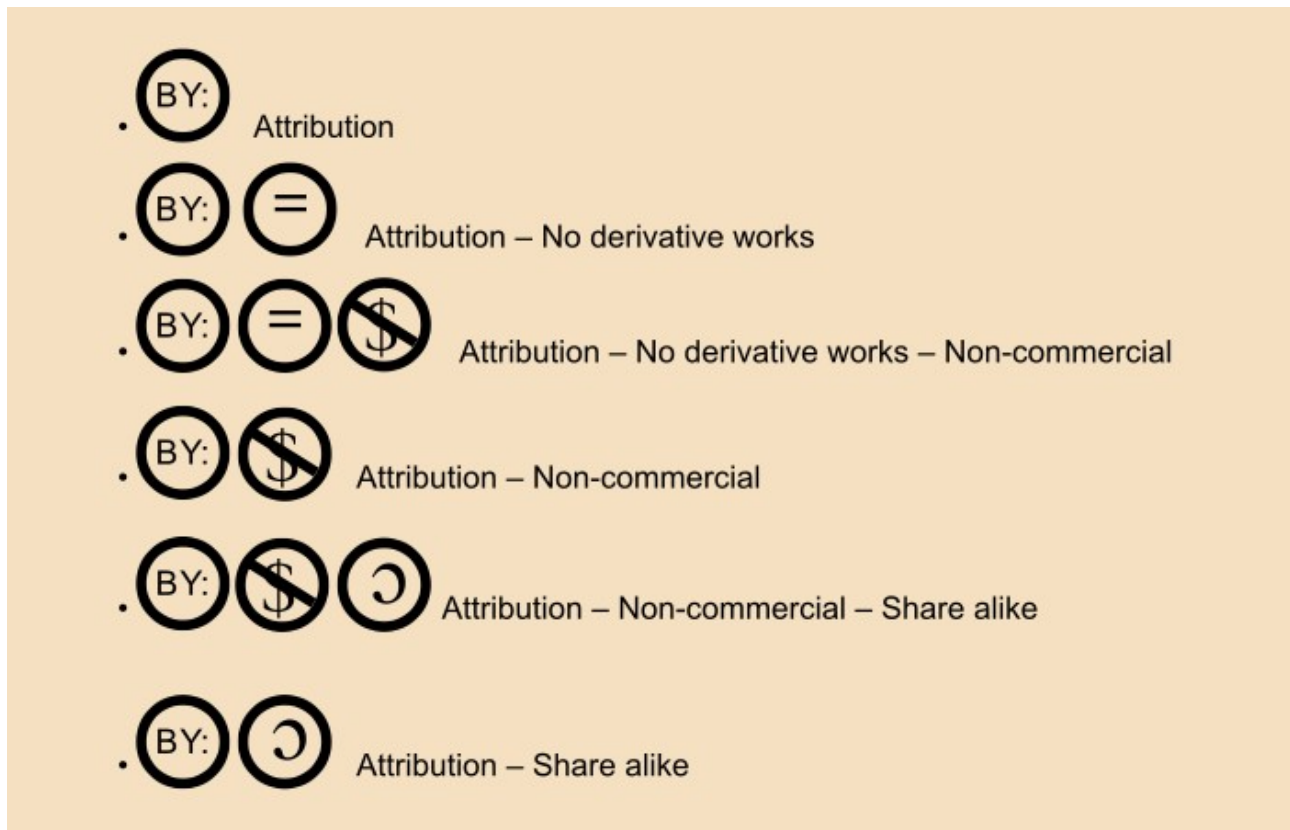
- αρκετά ισχυρές, για να μπορούν αντέξουν την εξέταση από ένα δικαστήριο, σε πολλές χώρες.
- αρκετά απλές, για να μπορούν να χρησιμοποιούνται από μη νομομαθείς και μη δικηγόρους.
- αρκετά εξεζητημένες (sophisticated), για να μπορούν να ανιχνεύονται από τις διάφορες διαδικτυακές εφαρμογές.

Οι διάφορες άδειες επιτρέπουν στον δημιουργό να επιλέξει τι είδους ελευθερίες επιτρέπονται, εκτός από τη δυνατότητα αντιγραφής, σύμφωνα με τέσσερα βασικά σημεία:

-  or  Attribution. The material can be distributed, copied or exhibited by third parties as long as the original author is credited.
-  or  Non-commercial. The original material and derivative works can be distributed, copied or exhibited for non-commercial use.
-  No derivative works . The material can be distributed, copied or exhibited but may not be used to create new works deriving from the original.
-  Share alike. The material can be modified and distributed but under the same license terms as the original material.

Στην έκδοση 1.x των Αδειών Creative Commons, υπήρχαν 11 είδη αδείας, που συνδυαζόταν με τα τέσσερα βασικά χαρακτηριστικά που αναφέρονται παραπάνω. Το 98% των δημιουργών επέλεξαν

τη δυνατότητα "attribution". Κατά συνέπεια, από την έκδοση 2.x των Αδειών Creative Commons και εντεύθεν, η δυνατότητα "attribution" είναι ένα προαπαιτούμενο. Αυτό μειώνει τους αρχικώς 11 προβλεπόμενους τύπους αδείας, σε έξι, που είναι οι εξής:



Ο ακόλουθος πίνακας δείχνει ένα διάγραμμα των αδειών με τα αντίστοιχα σήματα/ εικονίδια. Αυτή η εικόνα είναι, συνήθως, ένα link για μία περίληψη της Αδείας, που φιλοξενείται στον ιστότοπο της Creative Commons [21].

	Allows modifications.	Allows modifications if shared alike.	Does not allow modifications.
Allows commercial use.			
Does not allow commercial use.			

Μπορεί να χρησιμοποιηθεί και η γενικής φύσεως εικόνα¹⁴, αντί για την εικόνα που αναπαριστά την



άδεια, αλλά θα πρέπει να συνδεθεί με την άδεια που επέλεξε ο συγγραφέας. Μπορεί κανείς να πάρει από την Creative Commons τον κώδικα HTML για το link με την Άδεια [21].

Μόλις επιλεγθεί η Άδεια και προστεθεί το αντίστοιχο εικονίδιο, τότε το έργο έχει αδειοδοτηθεί και θα λάβετε τα εξής:

- Την *Ιδρυτική Πράξη Commons (Commons deed)*. Μία περίληψη της Αδείας μαζί με τα αντίστοιχα εικονίδια της. Αυτή η περίληψη θα προβάλλεται όταν γίνεται κλικ πάνω στο link που θα πάρετε από την Creative Commons [21].
- Το *Νομικό Κώδικα*. Αυτό είναι το πλήρες νομικό κείμενο επί του οποίου στηρίζεται η Άδεια. Μπορεί κανείς να έχει πρόσβαση σε αυτό το κείμενο από την περίληψη που αναφέραμε πιο πάνω.
- Τον *Ψηφιακό Κώδικα*. Αυτή είναι η περιγραφή τύπου RDF (Resource Description Framework), που μπορούν να χρησιμοποιούν οι μηχανές αναζήτησης και άλλες εφαρμογές, για να εντοπίζουν το είδος της Αδείας και τους όρους χρήσης. (14)

Τον Φεβρουάριο του 2007, δημοσιεύθηκε η έκδοση 3.0 των Αδειών Creative Commons. Αυτή ήταν μία ενημέρωση που διόρθωσε πολλά από τα σφάλματα που εντόπισαν οι χρήστες της. Η πρώτη μεγάλη τροποποίηση είναι ότι η γενική άδεια δεν βασίζεται πλέον στο μοντέλο των ΗΠΑ, αλλά βασίζεται τώρα στην ορολογία της Συμβάσεως της Βέρνης. Δεύτερον, τα ηθικά δικαιώματα και οι Σύλλογοι διαχείρισης δικαιωμάτων, αναφέρονται ρητά, καθώς καταρτίστηκαν διαφορετικοί κανόνες για ην κάθε περίπτωση. Τρίτον και τελευταίο, τα κείμενα τόσο της *Ιδρυτικής πράξης Commons*, όσο και του *Νομικού Κώδικα* που συνόδευαν την κάθε Άδεια, τροποποιήθηκαν για να καταστήσουν πιο σαφή τη ρήτρα που προβλέπει ότι η αναγνώριση της ταυτότητας του συγγραφέα δεν δίνει το δικαίωμα στον αδειούχο να αφήσει να εννοείται ή να δίνει την εντύπωση ότι έχει μία σχέση ή ότι συνδέεται κατά οποιονδήποτε τρόπο με εκείνο που του παρέχει την Άδεια.

Επιπρόσθετα, η Creative Commons παρέχει άλλους τύπους Αδειών, για ειδικές εφαρμογές. Όπως:



-  **Public Domain**. License used to release the work from copyright completely.
-  **Developing Nations**. The most permissive license for countries considered to be in development by the World Bank.
-  **Sampling**. License used for sharing snippets (fragments of code which perform a useful function).
-  **Founders' Copyright**. License used to release the work from copyright after a period of fourteen or twenty-eight years.
-  **CC-GNU GPL**. A license which adds the Creative Commons' metadata and summary (Commons Deed) to the Free Software Foundation's GNU General Public License.
-  **GNU LGPL**. A license which adds the Creative Commons' metadata and summary (Commons Deed) to the Free Software Foundation's GNU General Public License.
-  **Wiki**. License for Wiki. In practical terms this is identical to the attribution and share alike license.
-  **Music Sharing**. License used to share music.

Από ορισμένους κύκλους που συνδέονται με το Ελεύθερο Λογισμικό, δεν θεωρούνται όλες οι Άδειες Creative Commons ως ελεύθερες, καθώς θα πρέπει να πληρούν τις τέσσερες βασικές ελευθερίες για να ονομαστούν “Ελεύθερες” (βλέπε κεφάλαιο 1.1.1) Ο Benjamin "Mako" Hill (developer του Debian και του Ubuntu) δημιούργησε τον ιστότοπο Freedomdefined.org (<http://freedomdefined.org/>) [28], με σκοπό να παράσχει έναν καλύτερο ορισμό σχετικά με το τι εστί Ελεύθερη κουλτούρα και τι όχι. Επ' αυτής της βάσεως, μόνο δύο από τις συνολικές έξι βασικές άδειες Creative Commons μπορούν να θεωρηθούν ως αυστηρώς Ελεύθερες: η *attribution alone* (BY) και η *attribution-share-alike* (BY-SA), από τις οποίες η τελευταία είναι επίσης και *copyleft*.

Βιβλιογραφία

- [1] Aap Project: <http://www.a-a-p.org>
- [2] Ada Core Technologies: <http://www.gnat.com/>
- [3] Alcôve: <http://www.alcove.com>
- [4] Alcôve-Labs: <http://www.alcove-labs.org>
- [5] Alioth: <http://alioth.debian.org>
- [6] Anjuta: <http://www.anjuta.org>
- [7] The Apache Ant Project: <http://ant.apache.org>
- [8] Arch Revision Control System: <http://www.gnu.org/software/gnu-arch/>
- [9] artofcode LLC: <http://artofcode.com/>
- [10] Autoconf: <http://www.gnu.org/software/autoconf>
- [11] Barrapunto: <http://barrapunto.com>
- [12] Bazaar GPL Distributed Version Control Software: <http://bazaar-vcs.org/>
- [13] Berlios. The Open Source Mediator: <http://berlios.de>
- [14] Bitkeeper Source Management: <http://www.bitkeeper.com>
- [15] Bruce Perens: <http://perens.com/OpenStandards/Definition.html>
- [16] Caldera: <http://www.sco.com>
- [17] Cisco Enterprise Print System: <http://ceps.sourceforge.net/>
- [18] Code::blocks: <http://www.codeblocks.org>
- [19] Commons: <http://commons.wikimedia.org/>
- [20] Concurrent Version System: <http://ximbiot.com/cvs/>
- [21] Creative Commons. <http://creativecommons.org>
- [22] Directory of Open Access Journals: <http://www.doaj.org>
- [23] Eclipse - An Open Development Platform: <http://www.eclipse.org>
- [24] eCos: <http://sources.redhat.com/ecos/>
- [25] eCos license 2.0: <http://www.gnu.org/licenses/ecos-license.html>

- [26] First Monday. Peer Reviewed Journal on the Internet: <http://firstmonday.org>
- [27] Free Software Foundation: <http://www.fsf.org>
- [28] Freedom Defined (Free Cultural Works): <http://freedomdefined.org/>
- [29] Fundación Wu Ming: <http://www.wumingfoundation.com>
- [30] GForge: <http://gforge.org>
- [31] Gettext: <http://www.gnu.org/software/gettext>
- [32] GNU Automake: <http://www.gnu.org/software/automake>
- [33] GNU Emacs: <http://www.gnu.org/software/emacs/>
- [34] GNU Libc: <http://www.gnu.org/software/libc>
- [35] GNU Libtool: <http://www.gnu.org/software/libtool>
- [36] GNU Make: <http://www.gnu.org/software/make/make.html>
- [37] GNU Troff: <http://www.gnu.org/software/groff/groff.html>
- [38] "Have you seen these hackers?": <http://www.mozilla.org/MPL/missing.html>
- [39] "History of TeX": <http://www.math.utah.edu/software/plot79/tex/history.html>
- [40] IBM Public License Version 1.0: <http://opensource.org/licenses/ibmpl.php>
- [41] Jam Product Information: <http://www.perforce.com/jam/jam.html>
- [42] KDevelop: <http://www.kdevelop.org>
- [43] Launchpad: <https://launchpad.net>
- [44] The Linux Documentation Project: <http://www.tldp.org>
- [45] LinuxCare: <http://www.levanta.com>
- [46] Mailman, the GNU Mailing List Manager: <http://www.list.org>
- [47] The Malone Bug Tracker: <https://launchpad.net/products/malone>
- [48] Metawiki: <http://meta.wikimedia.org/>
- [49] Mozilla Public License 1.1: <http://www.mozilla.org/MPL/MPL-1.1.html>
- [50] Mozilla Tinderbox: <http://www.mozilla.org/tinderbox.html>
- [51] NetBeans: <http://www.netbeans.org>
- [52] Open Cores: <http://www.opencores.org>
- [53] Open Directory Project Social Contract:
- [54] Open Source Initiative: <http://www.opensource.org>
- [55] Public Library of Science: <http://www.publiclibraryofscience.org>
- [56] Red Hat: <http://www.redhat.com>
- [57] Savannah: <http://savannah.gnu.org> and <http://savannah.nongnu.org>
- [58] Slashdot: News for Nerds. <http://slashdot.org>
- [59] Sleepycat License: <http://www.sleepycat.com/download/oslicense.html>

- [60] Sleepycat Software: <http://www.sleepycat.com/>
- [61] SourceForge: Open Source Software Development Website: <http://sourceforge.net>
- [62] Subversion: <http://subversion.tigris.org>
- [63] Texinfo - The GNU Documentation System:
- [64] Tigris.org: Open Source Software Engineering: <http://tigris.org>
- [65] W3c Document License:
<http://www.w3.org/Consortium/Legal/2002/copyright-documents-20021231>
- [66] Wiktionary: <http://www.wiktionary.org>
- [67] Wikibooks: <http://www.wikibooks.org/>
- [68] Wikinews: <http://wikinews.org/>
- [69] Wikipedia: <http://www.wikipedia.org>
- [70] Wikiquote: <http://www.wikiquote.org>
- [71] Wikispecies: <http://species.wikimedia.org/>
- [72] Wikiversity: <http://wikiversity.org/>
- [73] X Window System Release 11 License: http://www.x.org/Downloads_terms.html
- [74] Ximian: <http://www.novell.com/linux/ximian.html>
- [75] Zope Corporation: <http://www.zope.com/>
- [76] Zope Public License 2.0: <http://www.zope.org/Resources/ZPL>
- [77] Law on Intellectual Property. Spanish Royal Legislative Decree 1/1996, of 12th April (April 1996):
- [78] Affero General Public License, 2002: <http://www.affero.org/oagpl.html>
- [79] Law on Intellectual Property. Spanish Law 23/2006, of 7th July (July 2006):
- [80] Flossimpact Study. Technical Report, European Commission, 2007: <http://flossimpact.eu>
- [81] ISO JTC 1/SC 34. Standard Generalised Markup Language (SGML, ISO 8879), 1986:
- [82] **Antoniades, I.; Samoladas, I.; Stamelos, I.; Bleris, G. L.** "Dynamical simulation models of the open source development process" En: Koch [157].
<http://www.wu-wien.ac.at/~koch/oss-book/>
- [83] **Bailey, E. C.** (1998). *Maximum RPM. Taking the Red Hat package manager to the limit.*
<http://rikers.org/rpmbook/>
- [84] **González Barahona, J. M.** (2000). "Software libre, monopolios y otras yerbas". Todo Linux (3). <http://sinetgy.org/~jgb/articulos/soft-libre-monopolios/>
- [85] **González Barahona, J. M.** (2002). "¿Qué se hace con mi dinero?". Todo Linux (17).
<http://sinetgy.org/~jgb/articulos/sobre-administracion/>
- [86] **González Barahona, J. M.; Robles, G.** Libre Software Engineering Web Site.

<http://libresoft.dat.escet.urjc.es/>

[87] **González Barahona, J. M.; Robles, G.** (2003, mayo). "Unmounting the *code god* assumption". En: *Proceedings of the Fourth International Conference on eXtreme Programming and Agile Processes in Software Engineering*. Genoa, Italy.

[88] **González Barahona, J. M.; Robles, G.; Ortuño Pérez, M. A.; Roderó Merino, L.; Centeno González, J.; Matellán Olivera, V.; Castro Barbero, E. M.; De las Heras Quirós, P.** "Anatomy of two GNU/Linux distributions". En: Koch [157].

<http://www.wai.wu-wien.ac.at/~koch/oss-book/>

[89] **Barnson, M. P.** *The Bugzilla guide*.

<http://www.bugzilla.org/docs214/html/index.html>

[90] **Baudis, P.** "Cogito manual page".

<http://www.kernel.org/pub/software/scm/cogito/docs/>

[91] **Bezroukov, N.** (1998, diciembre). "A second look at the cathedral and the bazaar". First Monday, 4(12).

http://www.firstmonday.org/issues/issue4_12/bezroukov/index.html

[92] **Bodnar, L.** (2003). "Linux distributions. Facts and figures".

<http://www.distrowatch.com/stats.php?section=packagemanagement>

[93] **Boehm, B. W.** (1981). *Software Engineering Economics*. Prentice Hall.

[94] **Bradner, S.** (1996, October). "The Internet standards process. Revision 3 (rfc 2026, bcp 9)".

<http://www.ietf.org/rfc/rfc2026.txt>

[95] **Cederqvist, P.**; GNU (1993). "CVS - concurrent versions system".

<http://www.gnu.org/manual/cvs/index.html>

[96] **Collins-Sussman, B.; Fitzpatrick, B. W.; Pilato, C. M.** (2004). *Version control with Subversion*. O'Reilly & Associates (<http://www.ora.com>).

<http://svnbook.red-bean.com/>

[97] **Cunningham, W.** "Wiki design principles".

[98] **Dachary, L.** (2001). "Savannah, the next generation".

<http://savannah.gnu.org/docs/savannah-plan.html>

[99] **Autonomous Government of Andalucía** (2003, March). Decree 72/2003, of 18th March, on Measures to Promote the Knowledge Society in Andalucía.

<http://www.andaluciajunta.es/SP/AJ/CDA/Ficheros/ArchivosPdf/DecretoConocimiento.pdf>

[100] **De Boor, A.** *Pmake. A tutorial*. <http://docs.freebsd.org/44doc/psd/12.make/paper.html>

[101] **De Icaza, M.** "The story of the GNOME Project".

<http://primates.ximian.com/~miguel/gnome-history.html>

[102] **Senate of the Republic of France**. Forum sur la proposition de loi tendant à généraliser dans l'administration l'usage d'Internet et de logiciels libres.

<http://www.senat.fr/consult/loglibre/index.htm>

[103] **De las Heras Quirós, P.; González Barahona, J. M.** (2000). "Iniciativas de las administraciones públicas en relación al software libre". *Bole. TIC, ASTIC magazine* (14).

[104] **Debian**. "Debian free software guidelines".

http://www.debian.org/social_contract.html#guidelines

[105] **Debian**. *Debian policy manual*. <http://www.debian.org/doc/debian-policy/>

[106] **Debian**. "Debian social contract". http://www.debian.org/social_contract.html

[107] **Schriftenreihe der KBSt** (2003, July). Leitfaden für die migration von basissoftwarekomponenten auf serverund arbeitsplatzsystemen. Technical report, Koordinierungs- und Beratungsstelle der Bundesregierung für Informationstechnik in der Bundesverwaltung (KBSt).

http://www.kbst.bund.de/download/mlf_v1_de.pdf

[108] **DiBona, C.; Ockman, S.; Stone, M.** (ed.) (1999). *Open sources. Voices from the open source revolution*. O'Reilly & Associates. <http://www.oreilly.com/catalog/opensource/>

[109] **Open Directory Project**. <http://dmoz.org>

[110] Ehrenkrantz, J. R. (2003, May). "Release management within open source projects". In: *Proceedings of the 3 Workshop on Open Source Software Engineering at the 25 International Conference on Software Engineering*. Portland, USA

[111] **European Council** (1991). Council Directive 91/250/CEE of 14th May 1991, on the legal protection of computer programs. <http://europa.eu.int/scadplus/leg/es/lvb/l26027.htm>

[112] **Feller, J.; Fitzgerald, B; Hissam, S.; Lakhani, K.** (ed.) (2003). *Making sense of the bazaar*. O'Reilly.

[113] **Fogel, K.; Bar, M.** (2001). *Open source code development with CVS* (2nd edition). Paraglyph Press. <http://cvsbook.red-bean.com>

[114] **Electronic Frontier Foundation**. Open Audio.

http://www.eff.org/IP/Open_licenses/eff_oal.html

[115] **Free Software Foundation**. GPLv3. <http://gplv3.fsf.org>

[116] **Free Software Foundation**. LGPLv3. First discussion draft.

<http://gplv3.fsf.org/pipermail/info-gplv3/2006-July/000008.html>

[117] **Free Software Foundation** (1985): "The GNU Manifesto". <http://www.gnu.org/philosophy/>

[118] **Free Software Foundation** (1991, junio). GNU General Public License, version 2.

<http://www.fsf.org/licenses/gpl.html>

[119] **Free Software Foundation** (1999, February). GNU Lesser General Public License,

version 2.1. <http://www.fsf.org/licenses/lgpl.html>

[120] **Free Software Foundation**. "Free software definition".

<http://www.gnu.org/philosophy/free-sw.html>

[121] **Free Software Foundation**. "Free licenses". <http://www.gnu.org/licenses/license-list.html>

[122] **Garbee, B.; Koptein, H.; Lohner, N.; Lowe, W.; Mitchell, B.; Murdock, I.; Schulze, M.; Small, C.** "A brief history of Debian". In the package: *Debian-history*.

[123] **Germán, D.** (2002, May). "The evolution of GNOME". In: *Proceedings of the 2 Workshop on Open Source Software Engineering at the 24 International Conference on Software Engineering*. Florida, USA

[124] **Germán, D.; Mockus, A.** (2003, May): "Automating the measurement of open source projects". In: *Proceedings of the 3 Workshop on Open Source Software Engineering at the 25 International Conference on Software Engineering*. Portland, USA

[125] **Ghosh, R. A.** (1998, March). "Cooking pot markets: an economic model for the trade in free goods and services on the Internet. *First Monday*, 3(3).

http://www.firstmonday.dk/issues/issue3_3/ghosh/index.html

[126] **Ghosh, R. A.; Glott, R.; Krieger, B.; Robles, G.** (2002). *Free/libre and open source software: Survey and study. Part iv: "Survey of developers"*.

http://www.infonomics.nl/FLOSS/report/FLOSS_Final4.pdf

[127] **Ghosh, R. A.; Prakash, V. V.** (2000, July). "The orbited free software survey". *First Monday*, 5(7). http://www.firstmonday.dk/issues/issue5_7/ghosh/index.html

[128] Godfrey, M. W.; Tu, Q. (2000, August). "Evolution in open source software. A case study". In: *Proceedings of the 2000 International Conference on Software Maintainance*.

[129] **González, J. A.** (2002, March). "Carta al congresista Villanueva".

<http://www.gnu.org.pe/mscarta.html>

[130] **Goossens, M.; Rahtz, S.** (1999). *The LaTeX Web Companion*. Addison Wesley.

[131] **Grad, B.** (2002, January-March). "A personal recollection: IBM's unbundling of software and services". In: *IEEE Annals of the History of Computing*, 24(1):64-71.

[132] **Working Group on Libre Software** (1999). "Free software / open source. Information society opportunities for Europe?". <http://eu.conecta.it/paper.pdf>

[133] **GrULIC**. "Legislation on the use of free software by the State".

<http://proposicion.org.ar/doc/referencias/index.html.es>

[134] **Hamerly, J.; Paquin, T.; Walton, S.** (1999). "Freeing the source. The story of Mozilla". <http://www.oreilly.com/catalog/opensources/book/netrev.html>

[135] **Hammel, M. J.** (1991, December). "The history of xfree86". *Linux Magazine*.

http://www.linux-mag.com/2001-12/xfree86_01.html

[136] **Harris, S.** (2001, August). *The Tao of IETF. A novice's guide to the Internet engineering task force* (RFC 3160, FYI 17). <http://www.ietf.org/rfc/rfc3160.txt>

[137] **Harrison, P.** (2002). "The rational street performer protocol".

<http://www.logarithmic.net/pfh/RSPP>

[138] **Hasan, R.** "History of Linux". <http://ragib.hypermart.net/linux/>

[139] **Hauben, M.; Hauben, R.** (1997). *Netizens. On the history and impact of Usenet and the Internet*. IEEE Computer Society Press.

[140] **Healy, K.; Schussman, A.** (2003, January). "The ecology of open source software development". <http://opensource.mit.edu/papers/healyschussman.pdf>

[141] **Hecker, F.** (1998, May). "Setting up shop. The business of open-source software".

<http://www.hecker.org/writings/setting-up-shop.html>

[142] **Hecker, F.** (1998). "Setting up shop. The business of open-source software".

<http://www.hecker.org/writings/setting-up-shop.html>

[143] **Hertel, G.; Niedner, S.; Herrmann, S.** (2003). "Motivation of software developers in open source projects. An Internet-based survey of contributors to the Linux kernel".

<http://opensource.mit.edu/papers/rp-hertelniednerherrmann.pdf>

[144] **Himanen, P.** (2001). *The hacker ethic and the spirit of the information age*. Random House.

<http://www.hackerethic.org>

[145] **Hunt, F.; Johnson, P.** (2002). "On the Pareto distribution of SourceForge projects. Technical report". Centre for Technology Management, Cambridge University Engineering Department, Mill Lane, Cambridge CB2 1RX.

<http://www-mmd.eng.cam.ac.uk/people/fhh10/Sourceforge/Sourceforge%20paper.pdf>

[146] **Open Source Initiative.** "History of the OSI". <http://www.opensource.org/docs/history.php>

[147] **Hamilton, J. R.** (US ambassador to Peru) (2002, June). "Carta al presidente del Congreso de la República". <http://www.gnu.org.pe/lobbyusa-congreso.html>

[148] **Jones, P.** (2000, May). "Brook's law and open source. The more the merrier?".

<http://www-106.ibm.com/developerworks/opensource/library/osmerrier.html?dwzone=opensource>

[149] **Jorgensen, N.** "Incremental and decentralized integration in FreeBSD". In: Feller et al.

[112]. <http://www.dat.ruc.dk/~nielsj/research/papers/bazaar-freebsd.pdf>

[150] **Brooks, F. P.** (1975). *The mythical man-month. Essays on software engineering*. Addison-Wesley.

[151] **Kalt, C.** (2000, April). "Internet relay chat: architecture (RFC 2810)".

<http://www.ietf.org/rfc/rfc2810.txt>

- [152] **Kelsey, J.; Schneier, B.** (1998, November). "The street performer protocol". In: *Third USENIX Workshop on Electronic Commerce Proceedings*. USENIX Press.
http://www.counterpane.com/street_performer.html
- [153] **Kelsey, J.; Schneier, B.** (1999, June). "The street performer protocol and digital copy-rights". *First Monday*, 4(6). http://www.firstmonday.dk/issues/issue4_6/kelsey/
- [154] **Kelty, C. M.** (2001, December). "Free software/free science". *First Monday*, 6(12).
http://firstmonday.org/issues/issue6_12/kelty/index.html
- [155] **Khatib, J.** "OpenIPCore Hardware General Public License".
http://www.opencores.org/OIPC/OHGPL_17.shtml
- [156] **Knuth, D.** (1989). *The TeXbook*. Addison Welsley.
- [157] **Koch, S.** (ed.) (2003). *Free/open source software development*. Idea Group Inc.
<http://www.wu-wien.ac.at/~koch/oss-book/>
- [158] **Koch, S.; Schneider, G.** (2000). "Results from software engineering research into open source development projects using public data". In: *Diskussionspapiere zum Tätigkeitsfeld Informationsverarbeitung und Informationswirtschaft, H.R. Hansen und W.H. Janko (Hrsg.), Nr. 22*, Wirtschaftsuniversität Wien.
- [159] **Kovács, G. L.; Drozdik, S.; Succi, G.; Zuliani, P.** (2004). "Open source software for the public administration". In: *Proceedings of the 6 International Workshop on Computer Science and Information Technologies* (CIST 2004). Budapest, Hungary.
- [160] **Krishnamurthy, S.** (2002, May). "Cave or community? An empirical examination of 100 mature open source projects". *First Monday*, 7(6).
http://www.firstmonday.dk/issues/issue7_6/krishnamurthy/index.html
- [161] **Laffitte; Trégouet; Cabanel** (1999). Proposition de loi numéro 495. Senate of the Republic of France. <http://www.senat.fr/consult/loglibre/texteloi.html>
- [162] **Laffitte; Trégouet; Cabanel** (2000). Proposition de loi numéro 117. Senate of the Republic of France. <http://www.senat.fr/consult/loglibre/texteloi.html>
- [163] **Lamport, L.** (1994). *LaTeX user's guide and reference manual* (2nd edition). Addison Welsley, Reading, Mass.
- [164] **Lancashire, D.** (2001, December). "Code, culture and cash. The fading altruism of open source development". *First Monday*, 6(12).
http://www.firstmonday.dk/issues/issue6_12/lancashire/index.html
- [165] **Lehman, M. M.; Ramil, J. F; Wernick, P. D.** (1997, November). "Metrics and laws of software evolution. The nineties view". In: *Proceedings of the 4 International Symposium on Software Metrics*. <http://www.ece.utexas.edu/~perry/work/papers/feast1.pdf>

- [166] **Leiner, B. M.; Cerf, V. G.; Kahn, R. E.; Clark, D. D.; Kleinrock, L.; Lynch, D. C.; Postel, J.; Roberts, L. G.; Wolff, S.** (1997). "A brief history of the Internet". In: *Communications of the ACM*. <http://www.isoc.org/internet/history/brief.shtml>
- [167] **Netcraft Ltd.** August 2003 Web Server Survey, 2003. http://news.netcraft.com/archives/2003/08/01/august_2003_web_server_survey.html
- [168] **Lucovsky, M.** (2000). "From NT OS/2 to Windows 2000 and beyond. A software-engineering odyssey". http://www.usenix.org/events/usenix-win2000/invitedtalks/lucovsky_html/>
- [169] **McGraw, G.** "Building secure software: how to avoid security problems the right way". Cited by: David A. Wheeler in <http://www.dwheeler.com/sloc/>
- [170] **McKusick, M. K.** (1999). "Twenty years of Berkeley Unix. From AT&T owned to freely redistributable". In: DiBona et al. [108]. <http://www.oreilly.com/catalog/opensources/>
- [171] **SUN Microsystems** (2000). "Sun microsystems announces availability of StarOffice source code on OpenOffice.org". http://www.collab.net/news/press/2000/openoffice_live.html
- [172] **Mockus, A.; Fielding, R. T.; Herbsleb, J. D.** (2000, June). "A case study of open source software development: the Apache server". In: *Proceedings of the 22 International Conference on Software Engineering (ICSE 2000)*, pages 263272. Limerick, Ireland ACM Press.
- [173] **Molenaar, B.** "What is the context of charityware?". <http://www.moolenaar.net/Charityware.html>
- [174] **MIT OpenCourseWare.** <http://ocw.mit.edu>
- [175] **Nagel, L. W.** (1996, september). "The life of SPICE". In: 1996 *Bipolar Circuits and Technology Meeting*. Minneapolis, MN, US <http://www.icsl.ucla.edu/aagroup/Life%20of%20SPICE.html>
- [176] **Narduzzo, A.; Rossi, A.** (2003, May). "Modularity in action: GNU/Linux and free/ open source software development model unleashed". <http://opensource.mit.edu/papers/narduzzorossi.pdf>
- [177] **Newman, N.** (1999). "The origins and future of open source software". <http://www.netaction.org/opensrc/future/>
- [178] **Nupedia.** <http://www.nupedia.com>
- [179] **Villanueva Núñez, E.** (2002, April). "Letter to Microsoft Peru". <http://www.gnu.org.pe/rescon.html>
- [180] **Danish Board of Technology** (2002, October). "Open-source software in e-Government, analysis and recommendations drawn up by a working group under the danish board of technology. Technical report".
- [181] **Open Source Initiative.** "Open source licenses". <http://www.opensource.org/licenses/index.html>

- [182] **Pareto, W.** (1896). "Course of Political Economy". Lausanne.
- [183] **Perens, P.**; The Open Source Initiative (1998). "The open source definition".
http://www.opensource.org/docs/definition_plain.html
- [184] **GNU Peru.** "Proyectos ley de software libre en la Administración pública del Gobierno peruano, Congreso de la República". <http://www.gnu.org.pe/proleyap.html>
- [185] **Pinheiro, P.** (1999, December). Proposição pl-2269/1999: Dispõe sobre a utilização de programas abertos pelos entes de direito público e de direito privado sob controle acionário da administração pública. Câmara dos Deputados do Brasil.
http://www.camara.gov.br/Internet/sileg/Prop_Detalhe.asp?id=17879
<http://www.fenadados.org.br/software.htm>
- [186] **Pranevich, J.** (2003). "The wonderful world of Linux 2.6".
<http://www.kniggit.net/wwol26.html>
- [187] **The Debian Project.** "Debian developer map". <http://www.debian.org/devel/developers.loc>
- [188] **Puigcercós Boixassa, J.** (2002). Draft Bill on Measures for Implementing Free Software in Public Administration.
http://www.congreso.es/public_oficiales/L7/CONG/BOCG/B/B_244-01.PDF
- [189] **Quittner, J.; Slatalla, M.** (1998). *Speeding the net: the inside story of Netscape and how it challenged Microsoft*. Atlantic Monthly Pr.
- [190] **Rasch, C.** "A brief history of free/open source software movement".
<http://www.openknowledge.org/writing/open-source/scb/brief-open-source-history.html>
- [191] **Rasch, C.** (2001, May). "The Wall Street performer protocol. Using software completion bonds to fund open source software development". *First Monday*, 6(6).
- [192] **Raymond, E. R.** (2001, January). *The cathedral and the bazaar. Musings on Linux and open source by an accidental revolutionary*. O'Reilly & Associates (<http://www.ora.com>).
<http://catb.org/~esr/writings/cathedral-bazaar/>
- [193] **Reis, C R.; De Mattos Fortes, R. P.** (2002, February). "An overview of the software engineering process and tools in the Mozilla Project".
<http://opensource.mit.edu/papers/reismozilla.pdf>
- [194] **Rideau, F. R.** (2000). "Patents are an economic absurdity".
<http://fare.tunes.org/articles/patents.html>
- [195] **Roberts, L.** (1978, November). "The evolution of packet switching". *Proceedings of the IEEE*, (66).
- [196] **Robles, G.; González Barahona, J. M.; Centeno González, J.; Matellán Olivera, V.; Roderio Merino, L.** (2003, May). "Studying the evolution of libre software projects using publicly

available data". In: *Proceedings of the 3 Workshop on Open Source Software Engineering at the 25 International Conference on Software Engineering*. Portland, US.

[197] **Robles, G.; Scheider, H.; Tretkowski, I.; Weber, N.** (2001): "Who is doing it? Knowing more about libre software developers". <http://widi.berlios.de/paper/study.pdf>

[198] **Rochkind, M.** (1986, May). "Interview with Dick Haight". *Unix Review*.

[199] **Scacchi, W.** (2003). "Understanding open source software evolution. Applying, breaking and rethinking the laws of software evolution".

<http://www.ics.uci.edu/~wscacchi/Papers/New/Understanding-OSS-Evolution.pdf>

[200] Schneier, B. (2000). "Software complexity and security".

<http://www.counterpane.com/crypto-gram-0003.html>

[201] **Smoogen, S. J.** "The truth behind Red Hat names".

http://www.smoogespace.com/documents/behind_the_names.html

[202] **Haggen So.** "Comparison of free/open source hosting (FOSPhost) sites available for hosting projects externally from project owners". <http://www.ibiblio.org/fosphost/exhost.htm>

[203] **Stallman, R.** "GNU coding standards". <http://www.gnu.org/prep/standards.html>

[204] **Stallman, R.** "*Why free software is better than open source*".

<http://www.fsf.org/philosophy/free-software-for-freedom.html>

[205] **Stallman, R.** (1998). "Copyleft: pragmatic idealism".

<http://www.gnu.org/philosophy/pragmatic.html>

[206] **Stallman, R.** (1998). "*Why free software is better than open source*".

<http://www.gnu.org/philosophy/free-software-for-freedom.html>

[207] **Stallman, R.** (1998). "Why software should not have owners".

<http://www.gnu.org/philosophy/why-free.html>

[208] **Stallman, R.** "The GNU Operating System and the Free Software Movement". In:

DiBona et al. [108]. <http://www.fsf.org/gnu/thegnuproject.html>

[209] **Stallman, R.** (1999, June). "On free hardware". *Linux Today*.

http://features.linuxtoday.com/news_story.php3?ltsn=1999-06-22-005-05-NW-LF

[210] **Stallman, R.** (2001). "The free universal encyclopedia and learning resource".

<http://www.gnu.org/encyclopedia/free-encyclopedia.html>

[211] **Stallman, R.** (2002). *Free software, free society. Selected essays of Richard M. Stallman*. Joshua Gay.

[212] **Stallman, R.** (2003). "Some confusing or loaded words and phrases that are worth avoiding". <http://www.gnu.org/philosophy/words-to-avoid.html>

[213] **Stoltz, M.** (1999). "The case for government promotion of open source software".

<http://www.netaction.org/opensrc/oss-report.html>

[214] **Tanenbaum, A.; Torvalds, L.** (1999). "The Tanenbaum-Torvalds debate".

<http://www.oreilly.com/catalog/opensources/book/appa.html>

[215] **The Open Source Initiative.** "The open source definition".

http://www.opensource.org/docs/definition_plain.html

[216] **Tiemann, M.** "Future of Cygnus Solutions. An entrepreneur's account". In: DiBona et al. [108].

<http://www.oreilly.com/catalog/opensources/book/tiemans.html>

[217] **Torvalds, L; Diamond; D.** (2001). *Just for fun: the story of an accidental revolutionary*. Texere.

[218] **Linus Torvalds, Hamano, J. C.; Ericsson, A.** "Git manual page".

<http://www.kernel.org/pub/software/scm/git/docs/>

[219] **Tuomi, I.** (2002). "Evolution of the Linux credits file: methodological challenges and reference data for open source research".

<http://www.jrc.es/~tuomiil/articles/EvolutionOfTheLinuxCreditsFile.pdf>

[220] **Several authors.** "Open letter to WIPO".

<http://www.cptech.org/ip/wipo/kamil-idris-7july2003.pdf>

[221] **Vigo i Sallent, P.; Benach i Pascual, E.; Huguet i Biosca; J.** (2002, May). Proposició de llei de programari lliure en el marc de l'Administració pública de Catalunya.

<http://www.parlament-cat.es/pdf/06b296.pdf>

<http://www.hispalinux.es/modules.php?>

[222] **Villanueva Núñez, E.** (2001, December). Free software project bill, number 1609.

<http://www.gnu.org.pe/proley1.html>

[223] **Villanueva Núñez, E.; Rodrich Ackerman, J.** (2002, April). Bill on the use of free software by the Public Administration, number 2485.

<http://www.gnu.org.pe/proley4.html>

[224] **W3C** (2000). *Extensible markup language (xml) 1.0* (2nd edition).

[225] **Walsh, N.; Muellner, L.; Stayton, B.** (2002). *DocBook: the definitive guide*. O'Reilly.

<http://docbook.org/tdg/en/html/docbook.html>

[226] **Welke, L; Johnson, L.** (1998). How the ICP Directory began.

<http://www.softwarehistory.org/history/Welke1.html>

[227] **Wheeler, D. A.** (2000, July). "Estimating Linux's size". <http://www.dwheeler.com/sloc>

[228] **Wheeler, D. A.** (2001, June). "More than a gigabuck: estimating GNU/Linux's".

<http://www.dwheeler.com/sloc>

- [229] **Wiesstein, E.** "Concise encyclopedia of mathematics". <http://mathworld.wolfram.com/>
- [230] **Wikipedia.** "Gini coefficient". http://www.wikipedia.org/wiki/Gini_coefficient
- [231] **Wikipedia.** "Lorenz curve". http://www.wikipedia.org/wiki/Lorenz_curve
- [232] **Wikipedia.** "Pareto". <http://www.wikipedia.org/wiki/Pareto>
- [233] **Wikipedia.** "TeX". <http://www.wikipedia.org/wiki/TeX>
- [234] **Wilson, B.** "Netscape Navigator". <http://www.blooberry.com/indexdot/history/netcape.htm>
- [235] **Computer World** (2000). "Salary survey 2000".
<http://www.computerworld.com/cwi/careers/surveysandreports>
- [236] **Young, R.** (1999). "Giving it away. how Red Hat software stumbled across a new economic model and helped improve an industry".
<http://www.oreilly.com/catalog/opensources/book/young.html>
- [237] **Zawinsky, J. W.** (1999). "Resignation and postmortem".
<http://www.jwz.org/gruntle/nomo.html>

Παραρτήματα

Jesús M. González Barahona
Joaquín Seoane Pascual
Gregorio Robles

PID_00148385



Universitat Oberta
de Catalunya

www.uoc.edu

Κατάλογος Περιεχομένων

1. Παράρτημα Α. Οδηγός εκμάθησης.....	5
2. Παράρτημα Β. Ημερομηνίες κλειδιά στην ιστορία του Ελεύθερου Λογισμικού.....	9
3. Παράρτημα Γ. Η GNU Public License	22
4. Παράρτημα Δ. Τα κείμενα μερικών προτάσεων Νόμου και τα σχετικά έγγραφα.....	25
5. Παράρτημα Ε. Creative Commons' Attribution-ShareAlike.....	56
6. Παράρτημα Ζ. Η Άδεια GNU Free Documentation License.....	64

1. Παράρτημα Α. Οδηγός εκμάθησης

A.1.Εισαγωγή

Τι είναι το Ελεύθερο Λογισμικό; Τι είναι η Άδεια ενός Ελεύθερου Προγράμματος και ποιές είναι οι επιπτώσεις της; Πώς αναπτύσσεται το Ελεύθερο Λογισμικό; Πώς χρηματοδοτούνται τα projects του Ελεύθερου Λογισμικού και με ποιιά επιχειρηματικά μοντέλα συνδέονται και τα οποία έχουμε εμπειρία; Τι δίνει κίνητρο στους developers, ειδικά στους εθελοντές, για να εμπλακούν στα projects Ελεύθερου Λογισμικού; Σαν πώς μοιάζουν οι αυτοί οι developers; Πώς συντονίζονται τα διάφορα projects που τρέχουν και με τι μοιάζει το λογισμικό που παράγουν; Κοντολογίς, ποιιά είναι η συνολική εικόνα του Ελεύθερου Λογισμικού;

Αυτά είναι τα ερωτήματα στα οποία θα προσπαθήσουμε να απαντήσουμε σε αυτό το κείμενο. Διότι, παρά την αυξανόμενη παρουσία του Ελεύθερου Λογισμικού στα ΜΜΕ και στις συζητήσεις μεταξύ των επαγγελματιών του τομέως ΙΤ, και έστω και αν οι απλοί πολίτες αρχίζουν γενικά να μιλούν γι' αυτό, παραμένει ακόμη άγνωστο σε πολλούς ανθρώπους. Ακόμη και εκείνοι που είναι εξοικειωμένοι με αυτό, συνειδητοποιούν μόνο κάποια από τα χαρακτηριστικά του, αλλά συνεχίζουν να αγνοούν κάποια άλλα.

A.2.Σκοποί

Ο γενικός σκοπός είναι, αναμφισβήτητα, να μπορέσει ο αναγνώστης να καταλάβει και να σκεφθεί λογικά για τις βασικές έννοιες το Ελεύθερου Λογισμικού και για τις κύριες επιπτώσεις τους. Ας προσπαθήσουμε να διερευνήσουμε κάποιους πιο εξειδικευμένους στόχους:

- Να γνωρίσουμε τι είναι (και τι δεν είναι) Ελεύθερο Λογισμικό και τις βασικές επιδράσεις που έχει ένας τέτοιος ορισμός.
- Να διερευνήσουμε τα στοιχειώδη νομικά θέματα που περιβάλλουν το Ελεύθερο Λογισμικό και, ιδιαίτερω, την σημασία των Αδειών, τους κύριους τύπους και τις επιπτώσεις τους.
- Να αποκτήσουμε μία αίσθηση της προοπτικής και της πραγματικότητας του Ελεύθερου Λογισμικού, από μία συνολική και ιστορική σκοπιά καθώς και από την πλευρά των πλέον προχωρημένων και σύγχρονων projects.
- Να μάθουμε και να γνωρίσουμε τις μεθόδους με τις οποίες τα projects Ελεύθερου Λογισμικού μπορούν να χρηματοδοτηθούν (όταν υφίστανται τέτοια μέσα) και τα κυριότερα επιχειρηματικά μοντέλα.
- Να μάθουμε τις σημαντικότερες λεπτομέρειες για τα μοντέλα ανάπτυξης του Ελεύθερου Λογισμικού και τις μεθόδους μελέτης τους, από την σκοπιά του software engineering.

A.3.Περιεχόμενα και Σχεδιασμός μαθημάτων

Αυτό το κείμενο είναι δομημένο σε διάφορα κεφάλαια (διδασκτικές ενότητες) και γραμμένα με τέτοιο τρόπο που να είναι πρακτικώς ανεξάρτητα και αυτόνομα, που σημαίνει ότι, με την εξαίρεση της εισαγωγής, το βιβλίο μπορεί να διαβασθεί με οποιαδήποτε σειρά. Συστήνουμε, πάντως, στους αναγνώστες να ακολουθήσουν την σειρά που προκαθορίζει το ίδιο το βιβλίο, σύμφωνα με το παρακάτω πλάνο.

Η σειρά μαθημάτων θα είναι δομημένη σε ECTS credits, που σημαίνει ότι ο σχεδιασμός θα απαιτήσει μία συνολική προσπάθεια εκ μέρους του μαθητή, που θα περιλαμβάνει ασκήσεις και αντιπαραθέσεις απόψεων (debates), που θα διαρκέσουν 150 ώρες.

Κεφάλαιο 1 (6 ώρες). Εισαγωγική ενότητα που εξετάζει όλες τις ειδικές πτυχές του Ελεύθερου Λογισμικού. Εστιάζει ουσιαστικά στο να δώσει μία εξήγηση για την υποκείμενη βάση, για εκείνους που μαθαίνουν σχετικά με τα θέματα αυτά για πρώτη φορά, αναδεικνύοντας τη σημασία του. Μία εισαγωγή που καλύπτει τον ορισμό του *Ελεύθερου Λογισμικού* και, μεταξύ άλλων, προσφέρει και την ανάλυση των κύριων επιπτώσεών του.

<i>Σκοποί</i>	<i>Περιεχόμενο</i>	<i>Υλικά</i>	<i>Δραστηριότητες</i>	<i>Χρόνος</i>
Να μάθουμε τι σημαίνει <i>ελευθερία</i> σχετικά με το Λογισμικό	Οι τέσσερις ελευθερίες	Κεφάλαιο 1.1.1	Ανάγνωση υλικού	2 ώρες
Να κάνουμε διάκριση ανάμεσα στο <i>Ελεύθερο Λογισμικό</i> και σε άλλες παρεμφερείς έννοιες	Ορισμός των συναφών εννοιών, είτε είναι ανάλογες, είτε παρόμοιες	Κεφάλαιο 1.1.2	Ανάγνωση υλικού και υποβολή εισηγήσεων	1 ώρα
Εισαγωγή στους λόγους για τους οποίους φτιάχνεται το Ελεύθερο Λογισμικό	Ηθικά και Πρακτικά κίνητρα	Κεφάλαιο 1.2	Ανάγνωση υλικού και υποβολή εισηγήσεων	1 ώρα
Εισαγωγή στις συνέπειες του Ελεύθερου Λογισμικού	Συνέπειες για τον χρήστη, το κράτος, τον developer, κλπ.	Κεφάλαιο 1.3	Ανάγνωση υλικού και υποβολή εισηγήσεων	2 ώρες

Κεφάλαιο 2 (14 ώρες). Η ιστορική ανάπτυξη του κόσμου του Ελεύθερου Λογισμικού, από την γέννησή της κατά τη δεκαετία του 1970, μέχρι και την παρούσα στιγμή, προσφέροντας μία ευρύτερη εικόνα των πλέον χαρακτηριστικών θεμέλιων λίθων της, των κύριων projects, της οικονομικής, επαγγελματικής και κοινωνικής εξέλιξης, κλπ.

<i>Σκοποί</i>	<i>Περιεχόμενο</i>	<i>Υλικά</i>	<i>Δραστηριότητες</i>	<i>Χρόνος</i>
Μαθαίνοντας την “προϊστορία” του Ελεύθερου Λογισμικού	Γεγονότα πριν την ύπαρξη αυτής της έννοιας	Κεφάλαιο 2.1 και αρχή του Παραρτήματος Β	Ανάγνωση υλικού και υποβολή εισηγήσεων	2 ώρες
Μαθαίνοντας όλη την ιστορία του Ελεύθερου Λογισμικού, μέχρι και σήμερα	Τα πιο σημαντικά γεγονότα κατά χρονολογική σειρά	Κεφάλαια 2.2, 2.3, 2.4 και το υπόλοιπο του Παραρτήματος Β	Ανάγνωση υλικού και υποβολή εισηγήσεων	10 ώρες
Προσπαθώντας να προβλέψουμε το μέλλον	Μερικές προβλέψεις (ελπίδες και προβλήματα)	Κεφάλαιο 2.5	Ανάγνωση υλικού και υποβολή εισηγήσεων	2 ώρες

Κεφάλαιο 3 (9 ώρες). Νομικές πλευρές του Ελεύθερου Λογισμικού. Οι πιο διαδεδομένες άδειες Ελεύθερου Λογισμικού και τα αποτελέσματά τους στα επιχειρηματικά μοντέλα και στα μοντέλα ανάπτυξης, θα αναλυθούν διεξοδικά.

<i>Σκοποί</i>	<i>Περιεχόμενο</i>	<i>Υλικά</i>	<i>Δραστηριότητες</i>	<i>Χρόνος</i>
Μαθαίνοντας τις βασικές έννοιες της πνευματικής και της βιομηχανικής ιδιοκτησίας	Πνευματικά Δικαιώματα (Copyright), Πνευματική ιδιοκτησία, Διπλώματα ευρεσιτεχνίας (patents), Εμπορικά σήματα (brands), Εμπορικά μυστικά	Κεφάλαιο 3.1	Ανάγνωση υλικού και υποβολή εισηγήσεων	3 ώρες
Μαθαίνοντας τη νομική βάση του Ελεύθερου Λογισμικού: τις Άδειες	Ορισμός των ελεύθερων αδειών και τα χαρακτηριστικά των σημαντικότερων αδειών	Κεφάλαιο 3.2	Ανάγνωση υλικού και υποβολή εισηγήσεων	7 ώρες

Κεφάλαιο 4 (8 ώρες). Χαρακτηριστικά των developers του Ελεύθερου Λογισμικού και τα κίνητρα που τους οδηγούν να συμμετέχουν στα διάφορα projects, καθιστώντας έτσι εφικτή την ίδια την ύπαρξη των ελεύθερων προγραμμάτων.

<i>Σκοποί</i>	<i>Περιεχόμενο</i>	<i>Υλικά</i>	<i>Δραστηριότητες</i>	<i>Χρόνος</i>
Γνωρίζοντας το είδος των ανθρώπων που αναπτύσσουν το Ελεύθερο Λογισμικό	Ηλικίες, φύλο, επάγγελμα, γεωγραφική κατανομή, κλπ.	Κεφάλαια 4.1, 4.2, 4.3 και 4.4	Ανάγνωση υλικού και υποβολή εισηγήσεων	4 ώρες
Μαθαίνοντας πόσο χρόνο αφιερώνουν σε αυτό και γιατί	Βαθμός αφιέρωσης σε εβδομαδιαία βάση, κίνητρα, θέματα κύρους και ηγεσίας	Κεφάλαια 4.5, 4.6, 4.7 και 4.8	Ανάγνωση υλικού και υποβολή εισηγήσεων	4 ώρες

Κεφάλαιο 5 (22 ώρες). Οικονομικές πλευρές του Ελεύθερου Λογισμικού και, ειδικά, μέθοδοι χρηματοδότησης των projects, καθώς και επιχειρηματικά μοντέλα που διερευνώνται.

<i>Σκοποί</i>	<i>Περιεχόμενο</i>	<i>Υλικά</i>	<i>Δραστηριότητες</i>	<i>Χρόνος</i>
Μαθαίνοντας για τις πηγές χρηματοδότησης	Πηγές Χρηματοδότησης που χρησιμοποιούνται	Κεφάλαιο 5.1	Ανάγνωση υλικού και υποβολή εισηγήσεων	8 ώρες
Μαθαίνοντας πως να μπορεί να βγει κέρδος από το Ελεύθερο Λογισμικό	Επιχειρηματικά Μοντέλα	Κεφάλαια 5.2 και 5.3	Ανάγνωση υλικού και υποβολή εισηγήσεων	8 ώρες
Μαθαίνοντας για τις σχέσεις ανάμεσα στο Ελεύθερο Λογισμικό και τις μονοπωλιακές καταστάσεις που είναι τυπικές στη βιομηχανία λογισμικού	Μονοπώλια και Λογισμικό. Ο ρόλος του Ελεύθερου Λογισμικού	Κεφάλαια 5.1, 5.2, 5.3 και 5.4	Ανάγνωση υλικού και υποβολή εισηγήσεων	6 ώρες

Κεφάλαιο 6 (28 ώρες). Σχέση ανάμεσα στις πολιτικές και το Ελεύθερο Λογισμικό και, ειδικά, ανάμεσα στις πολιτικές υπέρ της προώθησης του Ελεύθερου Λογισμικού και τη χρήση Ελεύθερου Λογισμικού εκ μέρους των Δημόσιων Οργανισμών.

<i>Σκοποί</i>	<i>Περιεχόμενο</i>	<i>Υλικά</i>	<i>Δραστηριότητες</i>	<i>Χρόνος</i>
Μαθαίνοντας για τις επιπτώσεις του Ελεύθερου Λογισμικού στους Δημόσιους Οργανισμούς.	Τα βασικά αποτελέσματα και οι δυσκολίες υλοποίησης	Κεφάλαιο 6.1	Ανάγνωση υλικού και υποβολή εισηγήσεων	4 ώρες
Μαθαίνοντας για το τι κάνουν (ή τι μπορούν να κάνουν) οι Δημόσιοι Οργανισμοί, σχετικά με το Ελεύθερο Λογισμικό	Λύσεις για τις ανάγκες, την προώθηση και την επένδυση σε Έρευνα & Ανάπτυξη	Κεφάλαιο 6.2	Ανάγνωση υλικού και υποβολή εισηγήσεων	4 ώρες
Μαθαίνοντας σχετικά με τις νομοθετικές πρωτοβουλίες	Ανασκόπηση νομοθετικών πρωτοβουλιών για την υλοποίηση ή την υποστήριξη Ελεύθερου Λογισμικού, περιλαμβάνοντας παραδείγματα ειδικών κειμένων.	Κεφάλαιο 6.3	Ανάγνωση υλικού και υποβολή εισηγήσεων	20 ώρες

Κεφάλαιο 7 (12 ώρες). Τα μοντέλα διαχείρισης και ανάπτυξης για projects Ελεύθερου Λογισμικού, οι τεχνικές που αποδείχθηκαν πετυχημένες και ποσοτικές καθώς και ποιοτικές μελέτες Ελεύθερου Λογισμικού από την άποψη της ανάπτυξης.

<i>Σκοποί</i>	<i>Περιεχόμενο</i>	<i>Υλικά</i>	<i>Δραστηριότητες</i>	<i>Χρόνος</i>
Μαθαίνοντας για τα παραδειγματικά μοντέλα ανάπτυξης λογισμικού	"The cathedral and the bazaar"	Κεφάλαια 7.1, 7.2, 7.3 και 7.5	Ανάγνωση υλικού και προτεινόμενης βιβλιογραφίας	3 ώρες
Μαθαίνοντας για τις διαδικασίες που εμπλέκονται στην ανάπτυξη Ελεύθερου Λογισμικού	Χαρακτηριστικές διαδικασίες	Κεφάλαιο 7.4	Ανάγνωση υλικού και προτεινόμενης βιβλιογραφίας	3 ώρες
Μαθαίνοντας για τις δυνατότητες και τις πραγματικότητες που η διαθεσιμότητα των πηγών και των σχετικών αρχείων φέρνουν στο engineering του Ελεύθερου Λογισμικού	Πόροι και ποσοτικές μελέτες	Κεφάλαιο 7.6	Ανάγνωση υλικού και προτεινόμενης βιβλιογραφίας	3 ώρες
Μαθαίνοντας για το τι απομένει να γίνει με το engineering Ελεύθερου Λογισμικού	Μελλοντικές εργασίες	Κεφάλαιο 7.7	Ανάγνωση υλικού και προτεινόμενης βιβλιογραφίας	3 ώρες

Κεφάλαιο 8 (14 ώρες). Εισαγωγή στις τεχνολογίες και τα περιβάλλοντα ανάπτυξης του Ελεύθερου Λογισμικού και τα αποτελέσματά τους στη διαχείριση και εξέλιξη των projects.

<i>Σκοποί</i>	<i>Περιεχόμενο</i>	<i>Υλικά</i>	<i>Δραστηριότητες</i>	<i>Χρόνος</i>
Μαθαίνοντας τα γενικά χαρακτηριστικά για τα περιβάλλοντα και τα εργαλεία που χρησιμοποιούν οι developers Ελεύθερου Λογισμικού	Γενική κατάταξη χαρακτηριστικών	Κεφάλαιο 8.1	Ανάγνωση υλικού και προτεινόμενης βιβλιογραφίας	1/2 ώρα
Μαθαίνοντας τα βασικά εργαλεία ανάπτυξης	Γλώσσες, μεταγλωττιστές λειτουργικά συστήματα, κλπ.	Κεφάλαια 8.2 και 8.3	Ανάγνωση υλικού και προτεινόμενης βιβλιογραφίας	2 ώρες
Μαθαίνοντας τις βασικές μεθόδους με τις οποίες οι developers συνεργάζονται	Ανταλλαγή μηνυμάτων, forums, αποθετήρια, chats και wikis	Κεφάλαιο 8.4	Ανάγνωση υλικού και προτεινόμενης βιβλιογραφίας	2 ώρες
Μαθαίνοντας τους μηχανισμούς που χρησιμοποιούνται για τη διαχείριση των πηγών και των εκδόσεων	CVS και νέες εναλλακτικές	Κεφάλαιο 8.5	Ανάγνωση υλικού και προτεινόμενης βιβλιογραφίας	4 ώρες
Μαθαίνοντας πως παράγεται η τεκμηρίωση για το Ελεύθερο Λογισμικό	Γλώσσες και εργαλεία για παραγωγή τεκμηρίωσης	Κεφάλαιο 8.6	Ανάγνωση υλικού και προτεινόμενης βιβλιογραφίας	2 ώρες
Μαθαίνοντας πως γίνεται η διαχείριση σφαλμάτων και των εργασιών	Σύστημα διαχείρισης Bugs	Κεφάλαιο 8.7	Ανάγνωση υλικού και προτεινόμενης βιβλιογραφίας	1 ώρα
Μαθαίνοντας πως υποστηρίζεται η φορητότητα	Πόροι για άλλες αρχιτεκτονικές	Κεφάλαιο 8.8	Ανάγνωση υλικού και προτεινόμενης βιβλιογραφίας	1/2 ώρα
Μαθαίνοντας για τα δημόσια περιβάλλοντα ολοκληρωμένης ανάπτυξης	Το SourceForge και οι άλλοι	Κεφάλαιο 8.9	Ανάγνωση υλικού και προτεινόμενης βιβλιογραφίας	2 ώρες

Κεφάλαιο 9 (30 ώρες). Μελετώντας τα projects Ελεύθερου Λογισμικού (ανασκοπώντας τα πιο κλασικά και ενδιαφέροντα projects Ελεύθερου Λογισμικού, από την άποψη των αποτελεσμάτων που επετεύχθησαν, του μοντέλου διαχείρισης, της ιστορικής εξέλιξης, του αποτελέσματος επί των άλλων projects, κλπ.). Μελέτη των εταιρειών που σχετίζονται με Ελεύθερο Λογισμικό.

<i>Σκοποί</i>	<i>Περιεχόμενο</i>	<i>Υλικά</i>	<i>Δραστηριότητες</i>	<i>Χρόνος</i>
Μαθαίνοντας ένα παράδειγμα λειτουργικού συστήματος	Linux και *BSD	Κεφάλαια 9.1 και 9.2	Ανάγνωση υλικού και προτεινόμενης βιβλιογραφίας	8 ώρες
Μαθαίνοντας ένα παράδειγμα περιβάλλοντος επιφάνειας εργασίας	Gnome και KDE	Κεφάλαια 9.3 και 9.4	Ανάγνωση υλικού και προτεινόμενης βιβλιογραφίας	8 ώρες
Μαθαίνοντας ένα παράδειγμα προγραμμάτων συστήματος	Apache	Κεφάλαιο 9.5	Ανάγνωση υλικού και προτεινόμενης βιβλιογραφίας	2 ώρες
Μαθαίνοντας ένα παράδειγμα προγραμμάτων για τον τελικό χρήστη	Mozilla και OpenOffice	Κεφάλαια 9.6 και 9.7	Ανάγνωση υλικού και προτεινόμενης βιβλιογραφίας	4 ώρες
Μαθαίνοντας ένα παράδειγμα μίας Διανομής	Red Hat και Debian	Κεφάλαια 9.8 και 9.9	Ανάγνωση υλικού και προτεινόμενης βιβλιογραφίας	8 ώρες

Κεφάλαιο 10 (6 ώρες). Ενότητα στην οποία παρουσιάζονται Ελεύθεροι Πόροι πέραν του λογισμικού: πρόκειται για Πόρους που δημιουργήθηκαν, εν μέρει, χάρη στο Ελεύθερο Λογισμικό και στο μοντέλο που αυτό προσέφερε.

<i>Σκοποί</i>	<i>Περιεχόμενο</i>	<i>Υλικά</i>	<i>Δραστηριότητες</i>	<i>Χρόνος</i>
Μαθαίνοντας για άλλους Ελεύθερους Πόρους	Ελεύθερα Κείμενα, hardware, διδακτικό υλικό και Τέχνη	Κεφάλαιο 10.1	Ανάγνωση υλικού και προτεινόμενης βιβλιογραφίας	3 ώρες
Μαθαίνοντας για το εφαρμόσιμο	Άδειες, ιδιαίτερα οι Άδειες Creative Commons	Κεφάλαιο 10.2	Ανάγνωση υλικού και προτεινόμενης βιβλιογραφίας	3 ώρες

2. Παράρτημα Β. Ημερομηνίες-κλειδιά στην ιστορία του Ελεύθερου Λογισμικού

Αυτός είναι, απλώς, ένας κατάλογος των ημερομηνιών που θα μπορούσαν να θεωρηθούν ως σημαντικές στην ιστορία του Ελεύθερου Λογισμικού. Βασίζεται σε εκείνον που εμφανίζεται στο [132] και εκείνον που παρέχει η Open Source Initiative [146] και δεν φιλοδοξεί να είναι αναλυτικός: υπάρχουν σίγουρα πολλές σημαντικές ημερομηνίες που δεν συμπεριλήφθηκαν στον κατάλογο. Ευελπιστούμε, παρ' όλα αυτά, να προσφέρουμε ένα αρκούντως πλήρες πανόραμα του ιστορικού τοπίου μέσα στο οποίο εξελίχθηκε το ελεύθερο Λογισμικό.

<i>Ημερομηνίες</i>	<i>Συμβάντα</i>
Δεκαετία του 1950 και του 1960	Το λογισμικό διανέμεται μαζί με τον πηγαίο του κώδικα, χωρίς κανένα περιορισμό σε ομάδες χρηστών, σαν το SHARE (της IBM) και το DECUS (της DEC).
Απρίλιος, 1969	Δημοσιεύεται το RFC υπ' αριθμ. 1, που περιγράφει το πρώτο Διαδίκτυο (τότε λεγόταν Arpanet). Η ελεύθερη διαθεσιμότητα των RFC και, ειδικά, των προδιαγραφών για τα πρωτόκολλα που χρησιμοποιούνταν στο Διαδίκτυο, αποτέλεσαν παράγοντες-κλειδιά στην ανάπτυξή του.
Ιανουάριος, 1970	Η IBM άρχισε να πωλεί χωριστά το λογισμικό της, κάνοντας την αρχή της βιομηχανίας του ιδιοταγούς λογισμικού.
1972	Το Unix αρχίζει να διαδίδεται στα Πανεπιστήμια και τα Ερευνητικά Κέντρα.
1973	Το Unix φθάνει στο Πανεπιστήμιο του Berkeley, στην California. Αρχίζει η ιστορία του Unix BSD.
1973	Το SPICE τίθεται από τον Donald O. Penderson σε δημόσια διάθεση. Συν τω χρόνω, θα γίνει το σημείο αναφοράς στο χώρο αυτό (προσομοιωτές ολοκληρωμένων κυκλωμάτων).
1978	Ο Donald Knuth, του Πανεπιστημίου του Stanford, αρχίζει να εργάζεται στο TeX, ένα σύστημα ηλεκτρονικής τυπογραφίας (typesetting) που θα διανέμεται ως ελεύθερο λογισμικό.
1983	Ο Richard Stallman γράφει το "The GNU Manifesto", στο οποίο ζητά την διαμοίραση του λογισμικού στο ευρύ κοινό.
1984	Αρχίζει το έργο GNU. Οι developers που εργάζονται σε αυτό, και που αρχικά συντονίζονται από τον Richard Stallman, αρχίζουν να δημιουργούν έναν μεγάλο αριθμό εργαλείων, που ήταν όμοια με εκείνα του Unix, περιλαμβανομένου και ενός editor (του Emacs) και ενός μεταγλωττιστή (του GCC). Ο σκοπός ήταν να χτισθεί ένα πλήρως ελεύθερο λειτουργικό σύστημα.
1985	Η Κοινοπραξία X, βασισμένη στο MIT, κυκλοφορεί το Σύστημα Παραθύρων X ως Ελεύθερο Λογισμικό, υπό μια Άδεια που δεν είναι παρά ελάχιστα περιοριστική.
1985	Ο Richard Stallman ιδρύει το Free Software Foundation. Ανάμεσα στις άλλες δραστηριότητες, το Ίδρυμα θα λειτουργήσει και ως κέντρο υποδοχής της χρηματοδότησης και των πόρων που θα βοηθήσουν την ανάπτυξη του GNU project, αλλά και ως ο ιδιοκτήτης της πνευματικής ιδιοκτησίας που απορρέει από το όλο project.
1989	Ιδρύεται η Cygnus, η πρώτη εταιρεία που, ουσιαστικά, παρέχει εμπορικές υπηρεσίες στο Ελεύθερο Λογισμικό (για υποστήριξη, ανάπτυξη και παραμετροποίηση Ελεύθερων προγραμμάτων).

<i>Ημερομηνίες</i>	<i>Συμβάντα</i>
1989	Ο Προσομοιωτής Δικτύου (Network Simulator, ή απλώς, NS) αρχίζει να αναπτύσσεται ως παραλλαγή του ΠΡΑΓΜΑΤΙΚΟΥ Network Simulator. Ο Ns είναι ένα Ελεύθερος προσομοιωτής δικτύου τηλεπικοινωνιών, που θα χρησιμοποιηθεί εκτεταμένα από Πανεπιστήμια σε όλο τον κόσμο και που θα γίνει, μέχρις ενός βαθμού, σημείο αναφοράς στο χώρο αυτό.
1990	Το Ίδρυμα Ελεύθερου Λογισμικού (Free Software Foundation) ανακοινώνει πως προτίθεται να κτίσει έναν πυρήνα που θα ονομάζεται GNU Hurd. Ο σκοπός αυτού του project είναι να ολοκληρώσει εκείνο που έλειπε περισσότερο απο την όλη στρατηγική του GNU project: ένα πλήρες Λειτουργικό Σύστημα.
1991	Ο William και ο Lynne Jolitz γράφουν στο <i>Dr. Dobbs Journal</i> μία σειρά για το πως να κάνει κανείς port το BSD Unix σε υπολογιστές βασισμένους στον i386.
Αύγουστος, 1991	Ο Linus Torvalds, ένας εικοσιενός ετών Φινλανδός φοιτητής, ανακοινώνει πως άρχισε να δουλεύει σε ένα ελεύθερο Unix-οειδή πυρήνα, χρησιμοποιώντας εργαλεία του έργου GNU, σαν τον GCC. Ο σκοπός του, τον καιρό εκείνο, ήταν να χτίσει μία ελεύθερη έκδοση του Minix.
Οκτώβριος, 1991	Ο Linus Torvalds κυκλοφορεί την πρώτη έκδοση του πυρήνα του, που είναι ακόμη πολύ πρωτόγονος και ονομάζεται Linux.
1992	Η Αμερικανική Πολεμική αεροπορία αναθέτει μία σύμβαση στο Πανεπιστήμιο της Νέας Υόρκης, για να φτιάξει έναν μεταγλωττιστή ανοικτού κώδικα, για τη νέα έκδοση της Ada (μία γλώσσα της οποίας η χρήση ήταν σχεδόν υποχρεωτική, τον καιρό εκείνο, σε όλες τις συμβάσεις με τον Αμερικανικό Στρατό), την Ada 95. Για τη δημιουργία του κώδικα, η ομάδα του NYU επέλεξε τον GNU GCC και τον αποκάλεσε Μεταγλωττιστή GNAT (GNU NYU Ada 95 Translator).
Ιούλιος, 1992	Ο William και ο Lynne Jolitz κυκλοφορούν το 386BSD 0.1, που, συν τω χρόνω, θα οδηγήσει στη δημιουργία των projects του NetBSD, του FreeBSD και αργότερα και του OpenBSD.
1993	Στη Γερμανία ιδρύεται το SuSE, που αρχίζει τη δραστηριότητά του διανέμοντας το Slackware Linux, μεταφρασμένο στα Γερμανικά.
Αύγουστος, 1993	Ο Ian Murdock αρχίζει μία νέα διανομή του Linux, με το όνομα Debian GNU/Linux, που θα γίνει μία διανομή χτισμένη από εθελοντές developers και με τους περισσότερους συμμετέχοντες.
Δεκέμβριος, 1993	Κυκλοφορεί στο Διαδίκτυο το FreeBSD 1.0, μία από τις πρώτες σταθερές διανομές που προέρχονται από το 386BSD, του Jolitz.

<i>Ημερομηνίες</i>	<i>Συμβάντα</i>
1994	Οι προγραμματιστές του GNAT ιδρύουν την εταιρεία Ada Core Technologies, με σκοπό να διασφαλίσουν την ανάπτυξή του και την εξέλιξή του στο μέλλον και με ένα επιχειρηματικό μοντέλο βασισμένο στη παροχή υπηρεσιών για τον μεταγλωττιστή των πελατών τους (και όχι πουλώντας αυτόν καθ' εαυτόν τον μεταγλωττιστή, που συνεχίζει να είναι Ελεύθερο Λογισμικό). Συν τω χρόνω, το GNAT θα γίνει ο ηγέτης στην αγορά των μεταγλωττιστών της Ada.
Ιανουάριος, 1994	Κυκλοφορεί η έκδοση 0.91 του Debian GNU/Linux. Είναι ο καρπός των προσπαθειών δώδεκα (12) developers.
Μάρτιος, 1994	Δημοσιεύεται η πρώτη έκδοση του <i>Linux Journal</i> .
29 Ιουλίου 1994	Ο Marc Ewing δημοσιεύει την πρώτη έκδοση του Red Hat Linux. Όπως και με το Debian, ο σκοπός είναι να βελτιώσει τα αποτελέσματα της κυρίαρχης, εκείνη την εποχή, διανομής, που ήταν το Slackware.
Οκτώβριος, 1994	Κυκλοφορεί το NetBSD 1.0.
1995	Ο Bob Young ιδρύει την Red Hat Software, μετά την εξαγορά της διανομής Red Hat Linux, από τον δημιουργό της, τον Marc Ewing, και τη συγχώνευσή της με τη δική του εταιρεία, την ACC, η οποία πωλούσε υλικό σχετικό με το Linux και το Unix, μέσω καταλόγου, από το 1993. Λίγο αργότερα, κυκλοφορεί η Red Hat Linux 2.0: είναι η πρώτη διανομή που περιλαμβάνει τη μορφή πακέτων RPM.
1995	Η DARPA υποστηρίζει την ανάπτυξη του NS μέσω του VINT project.
Ιανουάριος, 1995	Κυκλοφορεί το FreeBSD 2.0
Απρίλιος, 1995	Εμφανίζεται η πρώτη επίσημη release του Apache (0.6.2)
1996	Λαμβάνει χώρα η 1 ^η Διάσκεψη για το Ελεύθερα Αναδιανεμόμενο Λογισμικό, στο Cambridge της Μασσαχουσέτης, ΗΠΑ.
Οκτώβριος, 1996	Αναγγέλλεται το KDE project: είναι ένα από τα πρώτα που στοχεύει στην αντιμετώπιση των προβλημάτων χρηστικότητας στο περιβάλλον Unix, και το πρώτο που επιχειρεί κάτι τέτοιο σε μεγάλη κλίμακα, στον κόσμο του Ελεύθερου Λογισμικού.
Ιανουάριος, 1997	Ο Eric S. Raymond παρουσιάζει το έργο του "The cathedral and the bazaar", στο οποίο εκφράζει τις απόψεις του για ποιο λόγο ορισμένα αναπτυξιακά μοντέλα Ελεύθερου Λογισμικού δουλεύουν.

<i>Ημερομηνίες</i>	<i>Συμβάντα</i>
Αύγουστος, 1997	Ο Miguel de Icaza ανακοινώνει το GNOME project, έναν <i>ανταγωνιστή</i> του KDE με παρόμοιους σκοπούς, αλλά με τον δεδηλωμένο στόχο να διασφαλίσει ότι ολόκληρο το σύστημα που θα προκύψει θα είναι Ελεύθερο Λογισμικό. Προέκυψε σαν μία αντίδραση του Free Software Foundation και άλλων, προς το πρόβλημα αδειοδότησης που είχε το KDE, το οποίο συμπεριελάμβανε ένα βασικό συστατικό, την βιβλιοθήκη Qt, που δεν ήταν Ελεύθερο Λειτουργικό εκείνη την εποχή.
22 Ιανουαρίου 1998	Το Netscape δηλώνει την πρόθεσή του να διανείμει τον κώδικα του περιηγητή του (Navigator) ως Ελεύθερο Λογισμικό, που ήταν ο ηγέτης στην αγορά των περιηγητών διαδικτύου.
3 Φεβρουαρίου 1998	Οι Chris Peterson, Todd Anderson, John Hall, Larry Augustin, Sam Ockman και Eric Raymond, συναντιούνται για να μελετήσουν τις επιπτώσεις της αναγγελίας του Netscape για την κυκλοφορία του περιηγητή του και για να αποφασίσουν την προώθηση του όρου “Λογισμικό Ανοικτού Κώδικα” [146], χρησιμοποιώντας τον σαν εμπορικό λογότυπο που θα εγγυάται ότι τα προϊόντα που το έχουν, θα είναι <i>Ελεύθερο Λογισμικό</i> . Εκείνοι που προωθούν αυτόν τον όρο, αντιλαμβάνονται ότι είναι πιο κατάλληλος για τον εταιρικό κόσμο σε σχέση με εκείνον που χρησιμοποιείτο περισσότερο έως τότε, <i>Ελεύθερο Λογισμικό</i> . Δημιουργείται η Open Source Initiative για τη διαχείριση του όρου.
31 Μαρτίου 1998	Ο Netscape δίνει στη δημοσιότητα ένα μεγάλο τμήμα του πηγαίου του κώδικα για τον Netscape Navigator στο Διαδίκτυο.
7 Μαΐου 1998	Η Corel ανακοινώνει το NetWinder, έναν <i>δικτυακό υπολογιστή</i> βασισμένο στο Linux. Είναι η πρώτη φορά που μία μεγάλη εταιρεία εμπορεύεται κάτι που χρησιμοποιεί ένα λογισμικό που είναι βασικά Ελεύθερο Λογισμικό. Συντόμως μετά, η Corel ανακοινώνει το σχέδιο της να προσαρμόσει (port) το λογισμικό γραφείου (που περιλαμβάνει το WordPerfect) στο Linux, που αποτελεί επίσης μία καινοτομία για εκείνη την εποχή.
28 Μαΐου 1998	Η Sun Microsystems και η Adaptec γίνονται μέρος του Linux International. Είναι οι πρώτες μεγάλες εταιρείες IT που το πράττουν.
Ιούνιος, 1998	Το τεχνολογικό συνέδριο του USENIX, που συνήθως αφιερώνεται στο Unix, ανοίγει μία παράλληλη συνεδρία ονόματι FREENIX, που εστιάζει στο Ελεύθερο Λογισμικό.
22 Ιουνίου 1998	Η IBM ανακοινώνει ότι θα εμπορευθεί και θα προσφέρει στήριξη στο Apache, χρησιμοποιώντας το ως server για την γραμμή προϊόντων WebSphere.

<i>Ημερομηνίες</i>	<i>Συμβάντα</i>
Ιούλιος, 1998	Κυκλοφορεί το Debian GNU/Linux 2.0. Φτιάχτηκε από παραπάνω από 3000 εθελοντές και η διανομή περιλαμβάνει περισσότερα από 1500 πακέτα.
Ιούλιος, 1998	Κυκλοφορεί το KDE 1.0. Είναι η πρώτη σταθερή έκδοση που διανέμεται. Διάφορες διανομές GNU/Linux το ενσωματώνουν συντόμως μετά.
Αύγουστος, 1998	Ο Linus Torvalds και το Linux εμφανίζονται στο εξώφυλλο του περιοδικού Forbes.
Σεπτέμβριος, 1998	Η Red Hat, η δεσπόζουσα εταιρεία στην αγορά των διανομών που βασίζονται στο Linux την εποχή εκείνη, ανακοινώνει ότι η Intel και η Netscape αγόρασαν ένα μειοψηφικό ποσοστό μετοχών του κεφαλαίου της. Το Ελεύθερο Λογισμικό αρχίζει να ξυπνά το ενδιαφέρον των επενδυτών.
Νοέμβριος, 1998	Ιδρύεται η MandrakeSoft και συντόμως μετά, εκδίδει το Mandrake Linux, τη δική της διανομή GNU/Linux.
1 Νοεμβρίου 1998	Δημοσιεύονται τα έγγραφα Halloween, στα οποία η Microsoft φέρεται ότι εντοπίζει το GNU/Linux και το Ελεύθερο λογισμικό ως σημαντικούς ανταγωνιστές και σχεδιάζει πως να τα προσβάλλει.
27 Ιανουαρίου, 1999	Η εταιρεία HP και η SGI, ανακοινώνουν πως θα υποστηρίξουν το Linux στους υπολογιστές τους, πράγμα που σηματοδοτεί την ανάδυση μίας τάσης: την εγκατάλειψη του ιδιοταγούς Unix εκ μέρους των κατασκευαστών υπολογιστών που το είχαν ως λειτουργικό τους σύστημα, προς όφελος του Linux.
Μάρτιος, 1999	Κυκλοφορεί το GNOME 1.0, που στη συνέχεια θα γίνει πιο σταθερό (GNOME Οκτωβρίου) και θα ενσωματωθεί σε διάφορες διανομές GNU/Linux.
9 Μαρτίου 1999	Κυκλοφορεί το Debian GNU/Linux 2.1, με πάνω από 2000 πακέτα.
15 Μαρτίου 1999	Η Apple κυκλοφορεί το Darwin, που θα αποτελέσει το κεντρικό συστατικό του νέου της Mac OS X, υπό μία Ελεύθερη Άδεια.
Αύγουστος, 1999	Η Red Hat εισάγεται στο Χρηματιστήριο. Η τιμή των μετοχών της αυξάνεται τρομερά κατά τις πρώτες λίγες ημέρες μετά την εισαγωγή, σε βαθμό που η κεφαλαιακή της αξία αν έρχεται στα 4800 εκατομ. Δολάρια. Αργότερα, άλλες εταιρείες που σχετίζονται με το Ελεύθερο

<i>Ημερομηνίες</i>	<i>Συμβάντα</i>
	Λογισμικό, σαν την VA Linux και την Andover.net, θα εισηχθούν και αυτές στο Χρηματιστήριο. Η αξία των μετοχών όλων αυτών των εταιρειών θα κατακρημνισθεί μερικά χρόνια αργότερα, όταν σκάσει η φούσκα των dotcom. Πολλές από αυτές δεν θα επιβιώσουν από αυτό το συμβάν.
Οκτώβριος, 1999	Ιδρύονται δύο εταιρείες για να παράγουν λογισμικό, στα πλαίσια του GNOME project: η Eazel (που θα πτωχεύσει το 2002, αφού πρώτα δημιουργήσει το Nautilus, έναν διαχειριστή αρχείων) και η Helix Code (που μετονομάστηκε αργότερα σε Ximian και ύστερα εξαγοράστηκε από τη Novell, και που θα παρήγαγε εργαλεία σαν το Red Carpet ή το Evolution).
Νοέμβριος, 1999	Η Red Hat Software εξαγοράζει την Cygnus. Η εταιρεία που προκύπτει είναι η μεγαλύτερη στον κόσμο, στο χώρο του Ελεύθερου Λογισμικού.
Ιανουάριος, 2000	Κυκλοφορεί ο Mozilla M13, που θεωρήθηκε από πολλούς ως η πρώτη επαρκώς σταθερή έκδοση του Mozilla, σχεδόν 2 χρόνια μετά μετά τη κυκλοφορία μεγάλου τμήματος του κώδικα του Navigator .
Μάιος, 2000	Κυκλοφορεί το GNOME 1.2 (Bongo GNOME) .
Αύγουστος, 2000	Ανακοινώνεται η δημιουργία του Ιδρύματος GNOME Foundation.
15 Αυγούστου, 2000	Κυκλοφορεί το Debian GNU/Linux 2.2, με πάνω από 2500 πηγαία πακέτα, που περιλαμβάνουν περίπου 55 εκατομ. γραμμές κώδικα.
Ιανουάριος, 2001	Κυκλοφορεί η έκδοση 2.4 του πυρήνα Linux.
15 Ιανουαρίου 2001	Αρχίζει η Wikipedia. Η ιδέα της δημιουργίας μίας εγκυκλοπαίδειας χρησιμοποιώντας για υποστήριξη IT ένα wiki, όπου, επί της αρχής, μπορεί ο καθένας να συνεργασθεί, εφαρμόζοντας μεθόδους δουλειάς που μοιάζουν πολύ με εκείνες που χρησιμοποιούνται στο Ελεύθερο Λογισμικό, γίνεται πραγματικότητα.
30 Ιανουαρίου 2002	Ιδρύεται η ObjectWeb, μία οργάνωση που γεννήθηκε στη Γαλλία από τη Bull, την France Telecom και την INRIA, και είναι μία από τις πρώτες οργανώσεις σχεδιασμένες να παράγουν Ελεύθερο Λογισμικό, συνεργαζόμενη με εταιρείες και κέντρα έρευνας, με καθαρά εμπορικούς σκοπούς και με την ιδέα να καταστεί ο πυρήνας μίας διεθνούς κοινότητας συμφερόντων.

<i>Ημερομηνίες</i>	<i>Συμβάντα</i>
3 Απριλίου 2002	Κυκλοφορεί το KDE 3.0, η 3 ^η γενιά του περιβάλλοντος εργασίας KDE για την επιφάνεια εργασίας. Η ποιότητα των ελεύθερων επιφανειών εργασίας αρχίζει να αντιστοιχεί με εκείνη των παραδοσιακών εμπορικών επιφανειών εργασίας.
Απρίλιος, 2002	Ανακοινώνεται δημοσίως το project gnuLinEx. Με αυτό το project, η Διοίκηση της Περιφέρειας της Extremadura (Ισπανία) δηλώνει την πρόθεσή της να χρησιμοποιήσει μία διανομή GNU/Linux σε όλους τους υπολογιστές, όλων των δημοσίων Σχολείων της περιοχής.
Μάιος, 2002	Κυκλοφορεί το Mozilla 1.0, η πρώτη επισήμως σταθερή έκδοση του project.
1 Μαΐου 2002	Κυκλοφορεί η σουίτα γραφείου OpenOffice.org 1.0. Θα γίνει σύντομα η standard σουίτα εφαρμογών γραφείου στο κόσμο του Ελεύθερου Λογισμικού.
26 Ιουνίου 2002	Κυκλοφορεί το GNOME 2.0, που αντιπροσωπεύει ένα σημαντικό βήμα προς τα εμπρός για τους χρήστες, με μία πιο προσεκτικά σχεδιασμένη διεπαφή και περισσότερη προσοχή για τη φιλικότητα προς το χρήστη. Εισάγονται επίσης και άλλες πλευρές που βελτιώνουν την προσβασιμότητα.
19 Ιουλίου 2002	Κυκλοφορεί το Debian GNU/Linux 3.0, με πάνω από 100 εκατομ. γραμμών πηγαίου κώδικα. Στη έκδοση αυτή συμμετέχουν πάνω από 900 developers.
28 Ιουλίου 2002	Κυκλοφορεί η έκδοση 3.0 του Knoppix. Πρόκειται για μία δοκιμαστική διανομή (evaluation distribution) που μπορεί να εγκατασταθεί σε έναν σκληρό δίσκο πολύ γρήγορα, και σημειώνει μία τρομερή επιτυχία.
23 Σεπτεμβρίου 2002	Η 1 ^η έκδοση του Firefox (που εκείνη την εποχή λέγεται Phoenix) κυκλοφορεί σαν μία πειραματική επέκταση, βασισμένη στον κώδικα της σουίτας Mozilla, που υποτίθεται πως είναι απλούστερη.
Δεκέμβριος, 2002	Η Red Hat Software ανακοινώνει ότι οι ταμειακές της ροές, κατά το 2 ^ο και 3 ^ο τρίμηνο του 2002, ήταν θετικές.
16 Δεκεμβρίου 2002	Δημοσιεύονται οι πρώτες Άδειες Creative Commons (αν και το project ξεκίνησε το 2001).

<i>Ημερομηνίες</i>	<i>Συμβάντα</i>
Ιανουάριος, 2003	Η MandrakeSoft, μία εταιρεία που παράγει την διανομή Mandrake Linux, κηρύσσει πτώχευση.
19 Ιανουαρίου 2003	Κυκλοφορεί η έκδοση FreeBSD 5.0, μετά από σχεδόν 3 χρόνια δουλειάς από την προηγούμενη σταθερή και ευρείας κλίμακος έκδοση.
22 Ιανουαρίου 2003	Ο αριθμός των άρθρων στα Αγγλικά στη Wikipedia φθάνει τις 100.000. Λίγο μετά, ο αριθμός των Γερμανικών άρθρων φθάνει τις 10.000.
Φεβρουάριος, 2003	Η Motorola αρχίζει να πωλεί το A760 στην Κίνα. Είναι το πρώτο κινητό τηλέφωνο που διαθέτει ένα λειτουργικό σύστημα βασισμένο στο Linux (τη διανομή MontaVista Linux).
6 Μαρτίου 2003	Ο όμιλος SCO καταθέτει μία μήνυση κατά της IBM για την υποτίμηση της δικής της έκδοσης του Unix. Αυτό σηματοδοτεί την αρχή μίας αγωγής, στα πλαίσια της οποίας η IBM κατηγορήθηκε ότι συνεισέφερε κώδικα στον πυρήνα Linux, που όμως ανήκε στην SCO.
28 Μαΐου 2003	Ο Δήμος του Μονάχου, στη Γερμανία, ανακοινώνει ότι το Linux θα αντικαταστήσει τα Windows στα περισσότερα υπολογιστικά του συστήματα.
Ιούλιος, 2003	Η MandrakeSoft ανακοινώνει ότι ο ισολογισμός της ήταν θετικός για όλο το έτος και ότι προσδοκά να βγει από την επιτήρηση στα τέλη του 2003.
2003, 7 July	Μία ανοικτή επιστολή [220] απευθύνεται προς την WIPO (World Intellectual Property Organization), ζητώντας της να επανεξετάσει νέα ανοικτά μοντέλα συνεργατικής δημιουργίας (περιλαμβανομένου του Ελεύθερου Λογισμικού, αλλά και του project για το Ανθρώπινο Γονιδίωμα ή των Ανοικτών Επιστημονικών περιοδικών).
15 Ιουλίου 2003	Αρχίζει το Ίδρυμα Mozilla. Η Netscape Inc., τώρα ιδιοκτησία της AOL, αναγγέλλει ότι δε θα αναπτύξει περαιτέρω τον περιηγητή Netscape και, άρα, δε θα εργασθεί πλέον στο Mozilla project. Το Mozilla Foundation ιδρύθηκε με μία δωρεά 2.000.000 δολαρίων εκ μέρους της AOL και υλική υποστήριξη, καθώς και με ανθρώπινους πόρους από διάφορες εταιρείες, περιλαμβανομένης της ίδιας της AOL, της Red Hat και της Sun Microsystems.
4 Αυγούστου 2003	Η Novell εξαγοράζει τη Ximian Inc., μία από τις ηγετικές εταιρείες στην ανάπτυξη Ελεύθερου Λογισμικού (ειδικά για το GNOME), ως μέρος της στρατηγικής της να καταξιώσει τον εαυτό της στην αγορά των λύσεων σχετικών με το Linux.

<i>Ημερομηνίες</i>	<i>Συμβάντα</i>
2 Σεπτεμβρίου 2003	Κυκλοφορεί η έκδοση 1.1 του OpenOffice.org.
24 Σεπτεμβρίου 2003	Το Ευρωκοινοβούλιο τροποποιεί την Οδηγία για τη χορήγηση Διπλωμάτων ευρεσιτεχνίας (Patentability) στις εφευρέσεις που σχετίζονται με υπολογιστές, έτσι ώστε (αν γίνει δεκτή ως έχει) να μην επιτρέπονται τα διπλώματα ευρεσιτεχνίας στην Ε.Ε. Η Οδηγία, που απετέλεσε αρχικά πρόταση της Ευρωπαϊκής Επιτροπής, ακριβώς για να διασφαλισθεί ότι αυτού του είδους τα διπλώματα ευρεσιτεχνίας ήταν νόμιμα, βρίσκεται ακόμη στη διαδικασία συναπόφασης, κατά την οποία το Συμβούλιο των Υπουργών θα πρέπει επίσης να δώσει τη γνώμη του.
5 Νοεμβρίου 2003	Κυκλοφορεί η Έκδοση 1 (FC1) του Fedora Core, καρπός της διαδικασίας συνεργατικής ανάπτυξης που η Red Hat είχε αναγγείλει λίγους μήνες πριν. Από τη στιγμή εκείνη, η εταιρεία Red Hat θα πωλεί το Red Hat Enterprise Linux, ενώ οι συλλογές Fedora Core δεν συντηρούνται επισήμως από τη Red Hat, αλλά από την κοινότητα των εθελοντών developers, που την έφτιαξαν με τη βοήθεια της Red Hat (και που προϋπήρχε πριν αποφασίσει η Red Hat αυτή τη συνεργασία).
13 Ιανουαρίου 2004	Η Novell ολοκληρώνει την εξαγορά της SuSE, έναντι ενός συνολικού ποσού 210 εκατομ. δολαρίων.
9 Φεβρουαρίου 2004	Το Mozilla Foundation αποφασίζει να αλλάξει το όνομα του Mozilla Firebird (που πριν λεγόταν Phoenix) σε Mozilla Firefox. Αυτό θα παραμείνει και οριστικά ως όνομα του περιηγητή, ενώ η ανάπτυξή του πλησιάζει στην έκδοση 1.0.
18 Μαΐου 2004	Το Ευρωπαϊκό Συμβούλιο, στα πλαίσια της διαδικασίας συναποφάσεως για την Ευρωπαϊκή Οδηγία χορήγησης Διπλωμάτων Ευρεσιτεχνίας στις εφευρέσεις που σχετίζονται με υπολογιστές, αποφασίζει να υποβάλει μία συμβιβαστική έκδοση του κειμένου προς το Ευρωκοινοβούλιο. Κατηγορείται, πάντως, πως αγνόησε την ψήφο του Κοινοβουλίου, καθώς η νέα έκδοση επιτρέπει την έκδοση διπλωμάτων ευρεσιτεχνίας στο λογισμικό. Η απόφαση αμφισβητήθηκε τόσο πολύ, που ακόμη και μέσα στα πλαίσια του ίδιου του Συμβουλίου, δεν εγκρίθηκε παρά τον Μάρτιο του 2005.
8 Σεπτεμβρίου 2004	Η Pepper Computer αναγγέλλει ότι θα παρουσιάσει το πρώτο miniPC με οθόνη αφής, που χρησιμοποιεί ένα λειτουργικό σύστημα εντελώς ελεύθερο και βασισμένο στο Fedora Core.
20 Σεπτεμβρίου 2004	Ο αριθμός των άρθρων στη Wikipedia φθάνει στο ένα εκατομμύριο, σε 105 γλώσσες.

<i>Ημερομηνίες</i>	<i>Συμβάντα</i>
20 Οκτωβρίου 2004	Κυκλοφορεί η πρώτη έκδοση του Ubuntu. Βασίζεται στο Debian και ο σκοπός είναι να προσφέρει νέες εκδόσεις ανά τακτά διαστήματα. Το στήσιμο αυτής της διανομής χρηματοδοτείται από την εταιρεία Canonical, που παρέχει συντήρηση και υπηρεσίες προς τη διανομή. Αυτή η διανομή θα σημειώσει μεγάλη επιτυχία, σε μικρό χρονικό διάστημα.
9 Νοεμβρίου 2004	Κυκλοφορεί η έκδοση 1.0 του Firefox, μετά από μία μακρά σειρά προπαρασκευαστικών εκδόσεων. Την έκδοση αυτή την κατέβασαν παραπάνω από 25 εκατομ. φορές στις πρώτες 100 ημέρες μετά την κυκλοφορία.
24 Ιανουαρίου 2005	Η MandrakeSoft αναγγέλλει την εξαγορά της Βραζιλιάνικης εταιρείας Conectiva, που εκδίδει την ομώνυμη διανομή βασισμένη στο Linux. Λίγο μετά, η MandrakeSoft ανακοινώνει την αλλαγή του ονόματός της σε Mandriva.
1 Μαΐου 2005	Η OASIS αναγνωρίζει το ODF (Open Document Format), τη μορφή δεδομένων που χρησιμοποιεί το OpenOffice.org 2.0, μεταξύ άλλων, ως πρότυπο.
25 Μαΐου 2005	Η Nokia ανακοινώνει τη συσκευή Nokia 770, ένα mini-PC που χρησιμοποιεί μία έκδοση του Debian GNU/Linux, με το σύστημα X Window και το GTK+.
6 Ιουνίου 2005	Κυκλοφορεί το Debian GNU/Linux 3.1. Τώρα έχει πάνω από 200 εκατομ. γραμμές πηγαίου κώδικα.
14 Ιουνίου 2005	Η Sun Microsystems κυκλοφορεί το Open Solaris, την ελεύθερη έκδοση του λειτουργικού συστήματος Solaris.
15 Ιουνίου 2005	Η Mandriva εξαγοράζει την Αμερικανική εταιρεία Lycoris (τέως Redmond Linux) και αρχίζει να δουλεύει σε μια διανομή που ενσωματώνει τις προηγούμενες εκδόσεις της Mandrake, της Conectiva και της Lycoris.
6 Ιουλίου 2005	Το Ευρωκοινοβούλιο απορρίπτει, σε δεύτερη ανάγνωση, την πρόταση Οδηγίας για χορήγηση Διπλωμάτων Ευρεσιτεχνίας, σε εφευρέσεις που σχετίζονται με υπολογιστές, που προήλθε από το Συμβούλιο των Υπουργών. Αυτό σημαίνει ότι το μόνο νόμιμο κείμενο που μπορεί να εφαρμοσθεί στην Ε.Ε. είναι η Ευρωπαϊκή Σύμβαση για τα Διπλώματα Ευρεσιτεχνίας, του 1973.

<i>Ημερομηνίες</i>	<i>Συμβάντα</i>
20 Οκτωβρίου 2005	Κυκλοφορεί η έκδοση 2.0 του OpenOffice.org, που διανέμεται υπό την Άδεια LGPL.
Δεκέμβριος, 2005	Κυκλοφορεί η πρώτη έκδοση του Ruby on Rails, ενός περιβάλλοντος εργασίας για την ανάπτυξη διαδικτυακών εφαρμογών, με τη χρήση της αρχιτεκτονικής του model-view-controller. Δημοσιεύεται υπό την Άδεια X11, και θα χρησιμοποιηθεί ευρέως στο prototyping και την ανάπτυξη πολυάριθμων διαδικτυακών εφαρμογών.
Δεκέμβριος, 2005	Ο Nicholas Negroponte αναγγέλλει το project του OLPC (One Laptop Per Child), με τον σκοπό να σχεδιασθεί και να φτιαχθεί ένας φορητός PC των 100 δολαρίων, για τα παιδιά στις αναπτυσσόμενες χώρες. Χρησιμοποιεί Ελεύθερο Λογισμικό, με μία διανομή GNU/Linux ονόματι Sugar, που βασίζεται στη Red Hat.
14 Δεκεμβρίου 2005	Το επιστημονικό περιοδικό Nature δημοσιεύει ένα άρθρο που συγκρίνει την Wikipedia με την Βρετανική Εγκυκλοπαίδεια; σύμφωνα με το άρθρο, ο βαθμός ακρίβειας σε ό,τι αφορά τα επιστημονικά λήμματα, είναι παρόμοιος και στις δύο εγκυκλοπαίδειες.
16 Ιανουαρίου 2006	Δημοσιεύεται το πρώτο προσχέδιο της GPLv3: είναι μία προσπάθεια ενημέρωσης της GPL, που αποτελούσε, τον καιρό εκείνο, την πιο συχνά χρησιμοποιούμενη Άδεια για projects Ελεύθερου Λογισμικού (και με μεγάλη διαφορά). Σε αυτό το σημείο, αρχίζει μία διαδικασία ανοικτής αντιπαράθεσης γύρω από αυτές τις αλλαγές.
1 Μαρτίου 2006	Ο αριθμός των άρθρων στην Αγγλική γλώσσα της Wikipedia, φθάνει το ένα εκατομμύριο.
20 Μαρτίου 2006	Κυκλοφορεί το Fedora Core 5.
1 Ιουνίου 2006	Κυκλοφορεί το Ubuntu 6.06 LTS : διαφημίζεται η υποστήριξή του από την εταιρεία Canonical για τρία χρόνια.
Αύγουστος, 2006	Ο αριθμός των downloads του Firefox φθάνει τα 200 εκατομμύρια (υπάρχουν και πολλά ακόμη downloads από ανεπίσημους ιστότοπους, που δεν λαμβάνονται υπ' όψη). Σε αυτή τη χρονική περίοδο, υπολογίζεται ότι ο περιηγητής αυτός κατέχει μερίδιο της τάξης του 12% της συνολικής αγοράς (περίπου το 20% στην Ευρώπη).
12 Νοεμβρίου 2006	Η Sun ανακοινώνει ότι θα δώσει στην κυκλοφορία τις διάφορες εκδόσεις της πλατφόρμας Java υπό την Άδεια GPL. Μέχρι εκείνη τη στιγμή, αυτές οι εκδόσεις διανέμονταν δωρεάν σε εκτελέσιμη μορφή, πράγμα που η Sun δικαιολογούσε παραπέμποντας σε θέματα συμβατότητας και σταθερότητας. Αυτό, πάντως, έκανε εξαιρετικά δύσκολη την χρήση της Java στις διανομές Ελεύθερου Λογισμικού.

<i>Ημερομηνίες</i>	<i>Συμβάντα</i>
30 Νοεμβρίου 2006	Το ISO (Οργάνωση Διεθνών Προδιαγραφών/ International Standards Organization) και η IEC (International Electrotechnical Commission) δημοσιεύουν από κοινού την έκδοση ODF του OASIS, ως μία διεθνή προδιαγραφή (ISO/IEC 26300:2006) για την ανταλλαγή πληροφοριών γραφείου.
Δεκέμβριος, 2006	Η Ταϊβανέζικη εταιρεία First International Computer (FIC) παρουσιάζει το πρώτο προηγμένο κινητό τηλέφωνο που βασίζεται σε εντελώς Ανοικτό Κώδικα, κατά το Συνέδριο του Ανοικτού Κώδικα στην Κινητή. Ονομάζεται Neo1973, κοστίζει 350 δολάρια και χρησιμοποιεί μία πλατφόρμα λογισμικού που λέγεται OpenMoko, βασισμένη στον πυρήνα Linux 2.6, GTK+, X Windows και το Matchbox.
Ιανουάριος, 2007	Δημοσιεύεται η μελέτη FLOSSImpact [80], για τις επιπτώσεις (ειδικά τις οικονομικές) του Ελεύθερου Λογισμικού. Η μελέτη χρηματοδοτήθηκε από την Ευρωπαϊκή Επιτροπή και συνιστά την 1 ^η μελέτη ευρείας κλίμακος πάνω σε αυτό το θέμα.
23 Φεβρουαρίου 2007	Δημοσιεύεται η έκδοση 3.0 της Αδείας Creative Commons Licenses.
8 Απριλίου 2007	Κυκλοφορεί η έκδοση 4.0 του Debian GNU/Linux.

3. Παράρτημα Γ'. Η Δημόσια Άδεια GNU Public License

Έκδοση 2, Ιούνιος 1991

Copyright© 1989, 1991 Free Software Foundation, Inc. 675 Mass Ave, Cambridge, MA 02139, USA

Κυριολεκτικά αντίγραφα αυτού του κειμένου μπορούν να αντιγραφούν και να διανεμηθούν, αλλά δεν μπορούν να τροποποιηθούν.

Πρόλογος

Οι Άδειες για τα πιο πολλά λογισμικά έχουν σχεδιασθεί για να σου αφαιρούν την ελευθερία σου να μπορείς να μοιράζεσαι πράγματα και να τα αλλάζεις. Αντιθέτως, η Δημόσια Άδεια GNU General Public License έχει σκοπό να εγγυηθεί την ελευθερία σας να μπορείτε να μοιράζεστε και να τροποποιείτε το Ελεύθερο Λογισμικό, ούτως ώστε να διασφαλισθεί ότι το λογισμικό θα παραμένει ελεύθερο για όλους τους χρήστες του. Αυτή η Δημόσια Γενική Άδεια ισχύει για τα περισσότερα λογισμικά του Ιδρύματος Ελεύθερου Λογισμικού και για οποιοδήποτε άλλο πρόγραμμα του οποίου οι δημιουργοί δεσμεύονται στη χρήση της. (Μερικά άλλα λογισμικά του Ιδρύματος Ελεύθερου Λογισμικού καλύπτονται, αντιθέτως, από την Ελάχιστη Δημόσια Άδεια GNU Lesser General Public License.) Μπορείτε να την εφαρμόσετε και στα δικά σας προγράμματα.

Όταν μιλάμε για Ελεύθερο Λογισμικό, αναφερόμαστε στην ελευθερία, και όχι στο θέμα της τιμής. Οι δικές μας Γενικές Δημόσιες Άδειες (General Public Licenses) είναι σχεδιασμένες να εξασφαλίζουν ότι θα διατηρήσετε την ελευθερία σας να διανέμετε αντίγραφα Ελεύθερου Λογισμικού (ή να ζητάτε αμοιβή για την υπηρεσία αυτή, αν επιθυμείτε), να λαμβάνετε τον πηγαίο κώδικα ή και να τον πάρετε, αν θέλετε, να μπορείτε να τροποποιείτε το Λογισμικό ή να χρησιμοποιείτε τμήματά του σε νέα ελεύθερα προγράμματα και να γνωρίζετε ότι μπορείτε να κάνετε αυτά τα πράγματα.

Για να προστατέψουμε τα δικαιώματά σας, πρέπει να υιοθετήσουμε κάποιους περιορισμούς που απαγορεύουν σε οποιονδήποτε να σας αρνείται αυτά τα δικαιώματα ή να σας ζητά να τα αποποιηθείτε. Αυτοί οι περιορισμοί συνεπάγονται και κάποιες ευθύνες για εσάς, εάν διανέμετε αντίγραφα του λογισμικού, ή αν το τροποποιείτε.

Για παράδειγμα, αν διανείμετε αντίγραφα ενός τέτοιου προγράμματος, είτε πρόκειται για δωρεάν λογισμικό είτε όχι, θα πρέπει να αποδώσετε στους αποδέκτες όλα τα δικαιώματα που εσείς έχετε. Θα πρέπει να σιγουρευτείτε ότι και αυτοί, επίσης, λαμβάνουν ή ότι μπορούν να πάρουν τον πηγαίο κώδικα. Θα πρέπει, επίσης, να τους δείξετε αυτούς τους όρους, έτσι ώστε να γνωρίζουν τα δικαιώματά τους.

Προστατεύουμε τα δικαιώματά σας σε δύο βήματα:

Προστατεύουμε τα δικαιώματά σας σε δύο βήματα: (1) καλύπτοντας με πνευματικά δικαιώματα (copyright) το λογισμικό, και (2) προσφέροντάς σας αυτή την Άδεια η οποία σας δίνει τα νόμιμα δικαιώματα να αντιγράφετε, να διανέμετε, ή/και να τροποποιείτε το λογισμικό.

Επιπλέον, για την κάλυψη τόσο του κάθε συγγραφέως/δημιουργού, όσο και την δική μας, θέλουμε να βεβαιωθούμε ότι όλοι αντιλαμβάνονται πως δεν υφίσταται εγγύηση για αυτό το ελεύθερο λογισμικό. Αν το λογισμικό τροποποιηθεί από κάποιον και, κατόπιν, το περάσει σε κάποιον άλλον, τότε θέλουμε να γνωρίζουν οι παραλήπτες του ότι αυτό που έχουν δεν είναι το αρχικό, έτσι ώστε τα οποιαδήποτε προβλήματα που εισήγαγαν άλλοι, δεν θα έχουν αντίκτυπο στην υπόληψη και την φήμη των αρχικών συγγραφέων.

Τέλος, οποιοδήποτε ελεύθερο πρόγραμμα απειλείται μονίμως από τα διπλώματα ευρεσιτεχνίας λογισμικού (patents). Επιθυμούμε να αποφύγουμε τον κίνδυνο να προβαίνει ο κάθε μεμονωμένος αναδιανομέας ενός ελεύθερου προγράμματος σε έκδοση διπλωμάτων ευρεσιτεχνίας λογισμικού (patent licenses), μετατρέποντας έτσι το πρόγραμμα σε ιδιοταγές. Για να το αποφύγουμε αυτό, απαιτείται να ξεκαθαρίσουμε ότι είτε η οποιαδήποτε Άδεια πρέπει να παρέχει στον καθένα το δικαίωμα ελεύθερης χρήσης, είτε να μην χορηγείται καθόλου.

Ακολουθούν οι ακριβείς όροι και οι προϋποθέσεις για την αντιγραφή, αναδιανομή και την τροποποίηση:

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0) This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1) You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2) You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is

derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the program itself is interactive but does not normally print such an announcement, your work based on the program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the program.

In addition, mere aggregation of another work not based on the program with the program (or with a work based on the program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3) You may copy and distribute the program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable.

However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along

with the object code.

4) You may not copy, modify, sublicense, or distribute the program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5) You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the program (or any work based on the program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the program or works based on it.

6) Each time you redistribute the program (or any work based on the program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7) If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the program at all. For example, if a patent license would not permit royalty-free redistribution of the program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances. It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8) If the distribution and/or use of the program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9) The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10) If you wish to incorporate parts of the program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11) BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12) IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Πως να εφαρμόσετε αυτούς τους όρους στα καινούργια σας προγράμματα

Αν αναπτύξετε ένα νέο πρόγραμμα και θέλετε να έχει την ευρύτερη δυνατή χρησιμότητα για το κοινό, τότε ο καλύτερος τρόπος για να το πετύχετε είναι να το κυκλοφορήσετε ως Ελεύθερο Λογισμικό, το οποίο μπορεί ο καθένας να αναδιανέμει και να τροποποιεί, υπό αυτούς τους όρους. Για να το πράξετε, επισυνάψτε τις ακόλουθες προειδοποιήσεις στο πρόγραμμα. Είναι ασφαλέστερο να τις επισυνάψτε στην αρχή του κάθε πηγαίου αρχείου, για να κοινοποιείτε με τον αποτελεσματικότερο τρόπο την εξαίρεση της εγγυήσεως. Κάθε αρχείο θα πρέπει, επίσης, να έχει, τουλάχιστον, την αναφορά των πνευματικών δικαιωμάτων ("copyright" line) καθώς και την παραπομπή στο μέρος από όπου μπορεί κανείς να βρει την πλήρη προειδοποίηση:

one line to give the program's name and an idea of what it does. Copyright (C) yyyy name of author

Αυτό το πρόγραμμα αποτελεί Ελεύθερο Λογισμικό: μπορείτε να το αναδιανέμετε και/ή να το τροποποιείτε υπό τους όρους της Γενικής Δημόσιας Άδειας GNU General Public License, έτσι όπως δημοσιεύεται από το Ίδρυμα Ελεύθερου Λογισμικού, είτε υπό την έκδοση 2 της Άδειας, είτε (αναλόγως της επιθυμίας σας) υπό οποιαδήποτε μεταγενέστερη έκδοση.

Αυτό το πρόγραμμα διανέμεται με την ελπίδα ότι θα είναι χρήσιμο, αλλά ΧΩΡΙΣ ΚΑΜΙΑ ΕΓΓΥΗΣΗ, χωρίς ούτε την υπονοούμενη εγγύηση της ΕΜΠΟΡΕΥΣΙΜΟΤΗΤΑΣ ή της ΚΑΤΑΛΛΗΛΟΤΗΤΑΣ ΓΙΑ ΚΑΠΟΙΟΝ ΣΥΓΚΕΚΡΙΜΜΕΝΟ ΣΚΟΠΟ. Βλέπε την Γενική Δημόσια Άδεια GNU General Public License για παραπάνω πληροφορίες.

Θα πρέπει να έχετε λάβει ένα αντίγραφο της GNU General Public License μαζί με το πρόγραμμα. Αν όχι, γράψτε στο Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

Προσθέστε, επίσης, πληροφορίες σχετικά με το πώς να επικοινωνήσει κανείς μαζί σας μέσω ηλεκτρονικής αλληλογραφίας ή μέσω κλασικής ταχυδρομικής επιστολής.

Αν το πρόγραμμα είναι διαδραστικό, κάντε να βγάζει μία σύντομη γραμμή προειδοποίησης σαν κι αυτήν, όταν αρχίζει σε διαδραστικό mode:

Gnomovision version 69, Copyright (C) *yearname of author* Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

Οι υποθετικές εντολές 'show w' και 'show c' θα πρέπει να δείχνουν τα κατάλληλα τμήματα της Γενικής Δημόσιας Άδειας. Οι εντολές που χρησιμοποιείτε θα μπορούν, βεβαίως, να έχουν ένα διαφορετικό όνομα αντί για 'show w' και 'show c'. Θα μπορούσαν να είναι αντιστοιχούν ακόμη και σε κλικ του ποντικιού ή σε επιλογές του μενού (οτιδήποτε βολεύει το πρόγραμμά σας).

Θα πρέπει, επίσης, να βάλετε τον εργοδότη σας (αν εργάζεσθε σαν προγραμματιστής) ή το Σχολείο σας, αν υπάρχει, να υπογράψουν μία “αποκήρυξη πνευματικών δικαιωμάτων” (“copyright disclaimer”) για το πρόγραμμα, αν είναι απαραίτητο.

Ορίστε ένα παράδειγμα, αρκεί να αλλάξετε τα ονόματα:

Yoyodyne, Inc., δια της παρούσης αποποιείται και παραιτείται όλων των πνευματικών δικαιωμάτων στο πρόγραμμα 'Gnomovision' (το οποίο κάνει περάσματα σε μεταγλωττιστές) που γράφτηκε από τον James Hacker. Υπογράφεται από τον Ty Coon, 1 Απριλίου 1989, Ty Coon, Πρόεδρος της Vice

Αυτή η Γενική Δημόσια Άδεια δεν επιτρέπει την ενσωμάτωση του προγράμματός σας μέσα σε ιδιοταγή προγράμματα. Αν το πρόγραμμά σας είναι μία βιβλιοθήκη με υπορουτίνες (subroutine library), μπορεί να θεωρήσετε πιο χρήσιμο να επιτρέψετε σε ιδιοταγείς εφαρμογές να συνδεθούν με τη βιβλιοθήκη.

Αν, όντως, αυτό ακριβώς επιθυμείτε να πράξετε, τότε χρησιμοποιείστε την GNU Lesser General Public License, αντί για αυτή την Άδεια.

4. Παράρτημα Δ'. Κείμενα μερικών νομοσχεδίων και των σχετικών εγγράφων

Παρακάτω παρατίθεται το κυριολεκτικό κείμενο μερικών νομοσχεδίων που αναφέρθηκαν στο

κεφάλαιο 6 και μερικών σχετικών εγγράφων.

4.1. Νομοσχέδιο που παρουσίασαν οι Laffitte, Trégouët και Cabanel (Γαλλία)

Προσφέρουμε πιο κάτω μία μετάφραση του προταθέντος Νομοσχεδίου, που κατετέθη τον Οκτώβρη του 1999, από τους Γάλλους Γερουσιαστές Pierre Laffitte, René Trégouët και τον Guy Cabanel [laffitte99:_propos].

4.1.1. Recitals

(Περιλαμβάνονται μόνον οι παράγραφοι που αναφέρονται στο Ελεύθερο Λογισμικό.)

[...] Για να μπορέσει να εξασφαλισθεί η διαχρονική συνέχιση της προσβασιμότητας των δεδομένων, η διευκόλυνση της δυνατότητας ανταλλαγής τους καθώς και για να διασφαλισθεί ότι οι πολίτες θα έχουν ελεύθερη πρόσβαση στην πληροφορία, η χρήση αυτής της πληροφορίας εκ μέρους της Δημόσιας Διοίκησης δεν θα πρέπει να εξαρτάται από την καλή θέληση των κατασκευαστών λογισμικού. Είναι απαραίτητο να έχουμε ελεύθερα συστήματα, των οποίων η ανάπτυξη μπορεί να εξασφαλισθεί χάρη στο ότι ο πηγαίος κώδικας του κατασκευαστή θα είναι διαθέσιμος στους πάντες. Το Ελεύθερο Λογισμικό αυτή τη στιγμή αναπτύσσεται πολύ γρήγορα. Υπάρχουν πολλές εταιρείες IT που αναγνωρίζουν ότι το μέλλον της επιχείρησης δεν έγκειται στην πώληση του λογισμικού, αλλά στην παροχή υποστήριξης προς τους ανθρώπους που το χρησιμοποιούν, παρέχοντάς τους τις σχετικές υπηρεσίες.

Η δική μας πρόταση Νόμου θα καθορίζει ότι, μετά από μία μεταβατική περίοδο που θα διευκρινίζεται με Υπουργική απόφαση, η χρήση Ελεύθερου Λογισμικού θα καταστεί υποχρεωτική σε όλους τους Δημόσιους Οργανισμούς.

Το ιδιοταγές λογισμικό του οποίου ο πηγαίος κώδικας δεν διατίθεται ελεύθερα, θα μπορεί να χρησιμοποιείται μόνο σε συγκεκριμένες περιπτώσεις, όταν παρέχεται η σχετική εξουσιοδότηση από έναν φορέα ελεύθερου Λογισμικού. [...]

4.1.2. Άρθρα

- Άρθρο 1. Σχετικά με την απούλοποίηση (dematerialisation) της πληροφορίας και την ανταλλαγή δεδομένων μεταξύ των διαφόρων Δημοσίων Οργανισμών. Οι κρατικές υπηρεσίες, οι Τοπικές Αυτοδιοικήσεις και οι Δημόσιοι Οργανισμοί θα εξασφαλίζουν ότι όλες οι πληροφορίες και τα δεδομένα τους θα αποθηκεύονται σε ηλεκτρονική μορφή, μέσα σε ηλεκτρονικά δίκτυα, αρχής γενομένης από την 1^η Ιανουαρίου Άρθρο του 2002.

Οι όροι και οι συνθήκες που θα διέπουν την μετάβαση από την τρέχουσα χάρτινη μορφή ανταλλαγής εγγράφων και την μελλοντική ανταλλαγή που θα γίνεται με την χρήση ηλεκτρονικών μορφών αρχείων και τα δίκτυα, θα διευκρινισθεί μέσω Υπουργικού Διατάγματος.

- Άρθρο 2. Σχετικά με την απούλοποίηση (dematerialisation) των διαδικασιών δημόσιας αγοράς. Για να διασφαλισθεί ένας μεγάλος βαθμός διαφάνειας και ταχείας πρόσβασης στην πληροφορία για τις εταιρείες, όλες οι δημόσιοι διαγωνισμοί και τα σχετικά έγγραφα, θα δημοσιεύονται υπό ηλεκτρονική μορφή και σε δίκτυα, από την 1^η Ιανουαρίου 2002. Παρομοίως, όλες οι προσφορές που θα κατατίθενται σε διαγωνισμούς προμηθειών του Δημοσίου θα πρέπει να δημοσιεύονται υπό ηλεκτρον. μορφή και σε δίκτυα. Μία Υπουργική απόφαση θα καθορίζει τους μηχανισμούς της μετάβασης στην ηλεκτρονική διαδικασία.

• Άρθρο 3. Σχετικά με τις ανοικτές τεχνολογίες. Λαμβανομένων υπ' όψη των εξαιρέσεων που αναφέρονται στο άρθρο 4, και αρχής γενομένης από την 1^η Ιανουαρίου του 2002, οι κρατικές υπηρεσίες, οι Τοπικές Αυτοδιοικήσεις και οι Δημόσιοι Οργανισμοί θα μπορούν να χρησιμοποιούν μόνο Λογισμικό που διατίθεται ελεύθερα προς χρήση και τροποποίηση και για το οποίο ο πηγαίος κώδικας είναι διαθέσιμος.

Μία Υπουργική απόφαση θα καθορίζει τους όρους και τις προϋποθέσεις για την μετάβαση αυτή.

• Άρθρο 4. Σχετικά με τον Οργανισμό Ελεύθερου Λογισμικού. Θα δημιουργηθεί ένας Οργανισμός Ελεύθερου Λογισμικού. Θα είναι υπεύθυνος να πληροφορεί τις Κρατικές υπηρεσίες, τις Τοπικές Αυτοδιοικήσεις και τους Δημόσιους Οργανισμούς για τους όρους υπό τους οποίους θα πρέπει να εφαρμόζεται ο νόμος αυτός. Ο εν λόγω Οργανισμός θα καθορίζει και το είδος των Αδειών Λογισμικού που είναι κατάλληλες στα πλαίσια που προβλέπονται από τον νόμο αυτό.

Ο Οργανισμός αυτός θα εξασφαλίζει την διαλειτουργικότητα του Ελεύθερου Λογισμικού που θα χρησιμοποιείται στους Δημόσιους Οργανισμούς.

Ο Οργανισμός αυτός θα καταρτίσει έναν κατάλογο απογραφής, για κάθε τομέα, που θα περιλαμβάνει οποιαδήποτε αντικείμενο για το οποίο δεν υπάρχει διαθέσιμο Ελεύθερο Λογισμικό, ή δεν υπάρχει εφαρμόσιμο λογισμικό που να μπορεί να τροποποιηθεί και να χρησιμοποιείται ελεύθερα και του οποίου ο πηγαίος κώδικας διατίθεται ελεύθερα. Με βάση αυτή την καταγραφή, ο Οργανισμός θα ανακοινώσει τους αντίστοιχους Δημόσιους Οργανισμούς που θα μπορούν να εξαιρεθούν από τον Νόμο αυτό. Ο Οργανισμός ελεύθερου Λογισμικού θα είναι ανοικτός προς όλους τους χρήστες του Διαδικτύου, και θα πρέπει πριν από την κάθε τους απόφαση, να έχει προηγηθεί μία σχετική Διαδικτυακή διαβούλευση.

Θα ορισθεί ένας αντιπρόσωπος του Οργανισμού για το Ελεύθερο Λογισμικό σε κάθε Νομαρχία.

Ο τρόπος λειτουργίας του Οργανισμού για το Ελεύθερο Λογισμικό θα ορίζεται με Υπουργική απόφαση.

• Άρθρο 5. Σχετικά με την διάδοση των τροποποιήσεων που θα γίνουν στο Λογισμικό, στα πλαίσια του Νόμου αυτού.

Ο Οργανισμός Ελεύθερου Λογισμικού, ενώ θα τηρεί πλήρη σεβασμό των πνευματικών δικαιωμάτων, θα εξασφαλίζει ότι οι τροποποιήσεις στο λογισμικό θα διανέμονται σύμφωνα με το πλαίσιο που προβλέπεται από αυτό τον Νόμο.

• Άρθρο 6.

Τα κόστη που θα αναλάβει το κράτος ως συνέπεια του Νόμου αυτού θα αντισταθμισθούν μέσω αυξήσεων στα δικαιώματα που ορίζονται στα άρθρα 575 και 575Α του Γενικού Φορολογικού Νόμου.

4.2. Νομοσχέδιο των LeDéaut, Pauland Cohen (Γαλλία)

Θα προσφέρουμε τώρα μία μετάφραση, ουσιαστικά ολοκλήρου του προσχεδίου νόμου που παρουσιάστηκε από τους Jean-Yves Le Déaut, Christian Paul και Pierre Cohen, τον Απρίλιο του 2000.

4.2.1. Recitals

Η εκπληκτική ανάπτυξη στη χρήση των νέων τεχνολογιών της πληροφορίας και των τηλεπικοινωνιών, κατέστησε απαραίτητη την θέσπιση της σχετικής νομοθεσίας. Οι Δημόσιες Υπηρεσίες και οι Τοπικές Αυτοδιοικήσεις θα πρέπει να γίνουν το Πρότυπο και η ατμομηχανή για την Κοινωνία της Πληροφορίας, που θα εγγυάται τις ατομικές ελευθερίες, την ασφάλεια των καταναλωτών και την παροχή ίσων ευκαιριών σε αυτό τον τομέα.

Διάφορα παραδείγματα δείχνουν ότι, παρά την κάποια αξιολόγηση πρόοδο που επετεύχθη χάρη στις παρεμβάσεις της Κυβέρνησης στο χώρο της Κοινωνίας της Πληροφορίας, οι κρατικοί φορείς τείνουν να χρησιμοποιούν πρότυπα επικοινωνιών που συνδέονται στενά με ένα και μοναδικό ιδιωτικό προμηθευτή, πράγμα που σημαίνει ότι ένας χρήστης ή μία κοινότητα υποχρεώνονται να δράσουν ως πελάτες αυτού του ίδιου προμηθευτή, ενδυναμώνοντας κατ' αυτό τον τρόπο τα φαινόμενα κατάχρησης δεσπόζουσας θέσης στην αγορά.

Οι κρατικές υπηρεσίες χρησιμοποιούν συχνά λογισμικό του οποίου ο πηγαίος κώδικας δεν είναι διαθέσιμος, πράγμα που καθιστά αδύνατη την διόρθωση των διαφόρων bugs και των σφαλμάτων που οι ίδιοι οι προμηθευτές αρνούνται να διορθώσουν, είτε γίνεται αδύνατος ο έλεγχος των ενδεχομένων κενών ασφαλείας σε ευαίσθητες εφαρμογές. Οι Κρατικές υπηρεσίες χρησιμοποιούν, ενίοτε εν αγνοία τους, λογισμικό που μεταδίδει στα κρυφά πληροφορίες που είναι εξ' ορισμού εμπιστευτικές, προς ξένες κυβερνήσεις ή οργανώσεις.

Σε κάθε περίπτωση, τα οικονομικά μοντέλα της βιομηχανίας λογισμικού και τηλεπικοινωνιών που ανεπτύχθησαν στην αγορά, βασίζονται, σε μεγάλο βαθμό, στον σφετερισμό πελατείας και την εκθετική αξιολόγηση της απόκτησης των προφίλ των χρηστών. Αυτά τα οικονομικά μοντέλα επιβραβεύουν τις στρατηγικές παροχής ασύμβατων προϊόντων, τα βιομηχανικά μυστικά και την προγραμματισμένη απαρχαίωση, καθώς και την παραβίαση των ατομικών ελευθεριών. Αν και το Γαλλικό Κράτος δεν μπορεί να εξαφανίσει αυτές τις υποκείμενες τάσεις, διά της χρήσεως του Νόμου, λόγω της διασυννοριακής φύσεως των δικτύων επικοινωνιών, μπορεί, εν τούτοις, να διευκολύνει την ανάπτυξη μίας Κοινωνίας της Πληροφορίας επί Γαλλικού εδάφους, που να σέβεται τις κοινές δημόσιες ελευθερίες, την ασφάλεια των καταναλωτών και τις ίσες ευκαιρίες, και αυτό θα δημιουργούσε, ενδεχομένως, ένα προηγούμενο για την Ευρώπη και για όλο τον κόσμο.

Ο Νόμος βασίζεται σε πέντε αρχές:

- 1) το δικαίωμα του πολίτη να έχει ελεύθερη πρόσβαση στις δημόσιες πληροφορίες,
- 2) την διασφάλιση της διαχρονικής διατήρησης των δημόσιων δεδομένων,
- 3) την ασφάλεια του Κράτους,
- 4) την ασφάλεια των καταναλωτών στην Κοινωνία της Πληροφορίας και
- 5) την αρχή της διαλειτουργικότητας του λογισμικού.

Για να εξασφαλισθεί το δικαίωμα της ελεύθερης πρόσβασης του πολίτη στις δημόσιες πληροφορίες, ο κώδικας των ψηφιακών δεδομένων που παρέχονται από την Δημόσια Διοίκηση δεν πρέπει να συνδέεται με έναν και μοναδικό προμηθευτή. Τα Ανοικτά Πρότυπα, με άλλα λόγια, εκείνα για τα οποία οι κανονισμοί κωδικοποίησης των δεδομένων είναι δημόσιοι, καθιστούν εφικτή την παροχή εγγύησης ελεύθερης πρόσβασης, καθώς επιτρέπουν, όπου είναι απαραίτητο, την ανάπτυξη ελεύθερου και συμβατού λογισμικού.

Για να εξασφαλισθεί η διατηρησιμότητα των δημόσιων δεδομένων με το πέρασμα του χρόνου, η χρήση και η συντήρηση του λογισμικού δεν πρέπει να εξαρτάται από την καλή θέληση των δημιουργών του λογισμικού. Είναι επιβεβλημένο να έχουμε συστήματα των οποίων η ανάπτυξη είναι πάντοτε εξασφαλισμένη από την διαθεσιμότητα του πηγαίου κώδικα. Η αρχή της διαθεσιμότητας του πηγαίου κώδικα, στα πλαίσια των συμβάσεων που βασίζονται σε Άδειες, είναι μία αρχή που έως σήμερα υπήρχε μόνον ως επιλογή στη Νομοθεσία περί προμηθειών πακέτων λογισμικού και δημοσίων αγαθών (public utility), και θα πρέπει να γίνει ο κανόνας, εφαρμόζοντάς την σε όλες τις προκηρύξεις για προμήθειες του Δημόσιου Τομέα.

Θελήσαμε, ενσυνειδήτως, να αποφύγουμε μία διφορούμενη νομοθετική προσέγγιση που να

βασίζεται αποκλειστικά στη χρήση Ελεύθερου Λογισμικού. Δεν θα ήταν αρμόζον για το Κράτος, ανεξάρτητα από την αναγνωρισμένη ποιότητα του Ελεύθερου Λογισμικού, να παίρνει μέρος υπέρ ενός συγκεκριμένου οικονομικού μοντέλου για την κυκλοφορία λογισμικού. Αντιθέτως, η υποχρεωτική υιοθέτηση των Ανοικτών Προτύπων επικοινωνίας και η δημοσίευση του πηγαίου κώδικα, θα εξασφαλίσουν ίσες ευκαιρίες, σύμφωνα με τις αρχές της διαλειτουργικότητας που περιέχονται στη νομοθεσία περί λογισμικού.

Για να διαφυλαχθεί η εθνική ασφάλεια, είναι απαραίτητο να έχουμε συστήματα που να μη διαθέτουν στοιχεία που θα μπορούσαν να προσφέρουν δυνατότητες για μία απομεμακρυσμένη πρόσβαση στο σύστημα ή την ακούσια αναμετάδοση πληροφοριών προς οποιουδήποτε τρίτους φορείς. Χρειαζόμαστε συστήματα των οποίων ο πηγαίος κώδικας να είναι ελεύθερα προσβάσιμος από το ευρύ κοινό, έτσι ώστε να μπορεί να εξετασθεί από τον μεγαλύτερο δυνατό αριθμό ανεξάρτητων ειδικών παγκοσμίως φήμης. Το σχέδιο Νόμου που προτείνουμε θα προσφέρει μεγαλύτερη ασφάλεια για το Κράτος, καθώς η πλήρης γνώση του πηγαίου κώδικα θα εξαφανίσει τον αυξανόμενο αριθμό λογισμικών που περιέχουν "backdoors".

Το σχέδιο Νόμου που προτείνουμε θα ενισχύσει επίσης και την ασφάλεια των καταναλωτών στην Κοινωνία της Πληροφορίας, καθώς θα δίνει την δυνατότητα ανάδυσης νέων προσφορών για λογισμικά χωρίς "κερκόπορτες" ("backdoors"), που δε θα απειλούν την ιδιωτική ζωή και τις ατομικές ελευθερίες. Για να υπάρξουν, όμως, αληθινά ίσες ευκαιρίες, θα είναι απαραίτητο να τονισθεί εκ νέου και να εφαρμοσθεί η αρχή της διαλειτουργικότητας μέσα στη Νομοθεσία περί λογισμικού, και στη Νομοθεσία περί συμβατότητας. Σήμερα, και τα δύο αυτά δίκαια απειλούνται από εκείνους ακριβώς των οποίων η κυρίαρχη μονοπωλιακή θέση στην αγορά απειλείται, και οι οποίοι εγείρουν, συνεχώς εμπόδια για να παρεμποδίσουν την ανάπτυξη οποιουδήποτε ανταγωνισμού.

Για να διασφαλισθεί η διαλειτουργικότητα του λογισμικού, τα πνευματικά ή τα βιομηχανικά δικαιώματα ιδιοκτησίας (*intellectual or industrial property rights*) ενός δημιουργού λογισμικού, δεν θα πρέπει να εμποδίζουν την ανάπτυξη νέου συμβατού λογισμικού, που να μπορεί να το ανταγωνισθεί. Ο Νόμος πρέπει να εγγυάται το δικαίωμα στη συμβατότητα για όλους, με άλλα λόγια, το δικαίωμα για ελεύθερη ανάπτυξη, κυκλοφορία και χρήση πρωτότυπου λογισμικού, συμβατό με άλλα λογισμικά. Αντιστοίχως, η αρχή της διαλειτουργικότητας που εισήγαγαν οι Ευρωπαϊκοί Νόμοι περί λογισμικού, πρέπει να κυριαρχήσει επί των άλλων μορφών πνευματικής ή βιομηχανικής ιδιοκτησίας που ίσως ήταν σε ισχύ. ειδικότερα, η ύπαρξη μίας μάρκας για ένα πρότυπο επικοινωνίας, ή ενός διπλώματος ευρεσιτεχνίας για μία βιομηχανική διαδικασία που είναι απαραίτητη για την υλοποίηση του προτύπου επικοινωνίας, δεν πρέπει να επιτρέπει στον ιδιοκτήτη τους να εμποδίζει ή να περιορίζει την ελεύθερη διάδοση συμβατού Ελεύθερου Λογισμικού. Το Σχέδιο Νόμου που προτείνουμε μπορεί να εφαρμοσθεί άμεσα. Πραγματικά, οι περισσότεροι δημιουργοί λογισμικού είναι, ήδη, έτοιμοι να υιοθετήσουν Ανοικτά Πρότυπα επικοινωνίας, σαν εκείνα που ορίστηκαν στο Παρίσι, στη Βοστώνη και στο Τόκιο, από την Παγκόσμια Κοινοπραξία "World Wide Web Consortium".

Υπάρχουν πολλοί δημιουργοί ιδιοταγούς λογισμικού που είναι και αυτοί έτοιμοι να παραχωρήσουν στη Γαλλική Κυβέρνηση τον Πηγαίο Κώδικα των προϊόντων τους. Επιπλέον, η προσφορά Ελεύθερου Λογισμικού βασισμένου στο Λειτουργ. Σύστημα GNU/Linux, θα καλύψει πολλές από τις ανάγκες της Δημόσιας Διοίκησης, τόσο τώρα, όσο και στο μέλλον. Εν τούτοις, οι Δημόσιοι Οργανισμοί και τα σχετικά συλλογικά όργανα, δεν είναι επαρκώς ενημερωμένοι για την ύπαρξη των Ανοικτών Προτύπων ή την υπάρχουσα προσφορά λογισμικού μαζί με τον πηγαίο κώδικα.

Για να διευκολυνθεί η ταχεία υλοποίηση των Ανοικτών Προτύπων, είναι απαραίτητη η ενδυνάμωση

του ρόλου της Διυπουργικής Επιτροπής Τεχνικής Υποστήριξης για την Ανάπτυξη Τεχνολογιών της Πληροφορίας και της Επικοινωνίας εντός της Δημόσιας Διοίκησης (Mission Interministérielle de Soutien Technique pour le Développement des Technologies de l'Information et de la Communication dans l'Administration), και η επιφόρτισή της με την αποστολή να διεξαγάγει και να διαδώσει εντός της Δημόσιας Διοίκησης, μία απογραφή της υπάρχουσας προσφοράς Ανοικτών Προτύπων και Λογισμικού που να κυκλοφορεί μαζί με τον Πηγαίο του Κώδικα. Αν δεν διαπιστωθεί η ύπαρξη αγοράς γι' αυτό, τότε η παραπάνω Επιτροπή (MTIC) θα αναλάβει να αναπτύξει νέες προδιαγραφές ή νέα λογισμικά που θα δημοσιευθούν μαζί με τον Πηγαίο τους Κώδικα. Για να επιτελέσει αυτά τα νέα καθήκοντα, η Επιτροπή MTIC θα μετατραπεί σε μία Υπηρεσία Τεχνολογιών της Πληροφορίας και των Επικοινωνιών (Agency of Information Technologies and Communication, AITC).

Όταν δεν υπάρχει αγορά, το AITC θα αναλάβει να αναπτύξει νέες προδιαγραφές ή τα νέα λογισμικά που θα δημοσιεύονται με τον Πηγαίο τους Κώδικα. Για να διασφαλισθούν ίσες ευκαιρίες, η ανάπτυξη λογισμικών που θα υλοποιηθεί, θα τεθεί σε δημόσια διάθεση. Συνεπώς, τα λογισμικά που θα έχουν προκύψει με αυτόν τον τρόπο θα μπορούν να πωλούνται τόσο σαν ιδιοταγή, όσο και σαν Ελεύθερα Λογισμικά, αναλόγως του είδους της Αδείας που θα επιλεγεί ελεύθερα από τον Εκδότη. Η Υπηρεσία AITC θα είναι επίσης υπεύθυνη για την αποτίμηση του επιπέδου διαλειτουργικότητας, διαχρονικής διατηρησιμότητας και ασφάλειας του λογισμικού που θα προμηθεύεται η Γαλλική Δημόσια Διοίκηση.

Από μία γενικότερη άποψη, τα συστήματα ανοικτής επικοινωνίας και η διαθεσιμότητα του πηγαίου κώδικα αποτελούν βασικά στοιχεία για να εξασφαλισθεί η διαλειτουργικότητα, σε πανευρωπαϊκό επίπεδο, ανάμεσα στα διάφορα τοπικά συστήματα IT των κρατών-μελών και εκείνα των εθνικών οργανισμών, καθώς και για να αποφευχθεί η πιθανότητα εξάρτησης της διασύνδεσης των συστημάτων από την καλή θέληση και μόνον των δημιουργών του λογισμικού.

Η Υπηρεσία AITC θα έχει επίσης την αρμοδιότητα συμμετοχής σε projects διεθνούς συνεργασίας, στον τομέα των τεχνολογιών της πληροφορίας και των επικοινωνιών, καθώς επίσης και της διευκόλυνσης της διαλειτουργικότητας με τα πληροφορικά συστήματα των άλλων χωρών-μελών της Ε.Ε.

Το Σχέδιο Νόμου που προτείνουμε θα καλύπτει όλες τις παραπάνω ανησυχίες. Μας υπενθυμίζει ότι το Κράτος μπορεί να παίξει έναν σημαντικό ρόλο στην οικονομία, διαφυλάσσοντας τα εθνικά και τα Ευρωπαϊκά συμφέροντα, αλλά υπερασπιζόμενο και την οικονομία της αγοράς. Αυτό το Νομοσχέδιο θα επιτρέψει στη Γαλλία να προβληθεί ως ο υπερασπιστής της ελευθερίας, στα πλαίσια των νέων τεχνολογιών της πληροφορίας και των επικοινωνιών.

Δ.2.2.Άρθρα

• Άρθρο 1.

Η κρατική διοίκηση, οι τοπικές αυτοδιοικήσεις και όλα τα τοπικά σώματα, θα έχουν την υποχρέωση να χρησιμοποιούν τα Ανοικτά Πρότυπα επικοινωνίας για όλες τις ανταλλαγές ψηφιακών δεδομένων, σύμφωνα με όσα θα προβλέπουν οι δημόσιοι κανονισμοί και τα πρωτόκολλα σχετικά με την διαδικασία ανταλλαγής ψηφιακών δεδομένων.

• Άρθρο 2.

Η Δημόσια Διοίκηση, οι δημόσιοι οργανισμοί και τα κατά τόπους αυτοδιοικητικά σώματα, έχουν την υποχρέωση να χρησιμοποιούν λογισμικό του οποίου ο πηγαίος κώδικας να είναι διαθέσιμος.

- Άρθρο 3.

Όλα τα φυσικά πρόσωπα ή και οι εταιρικές οντότητες, έχουν το δικαίωμα να αναπτύσσουν, να δημοσιεύουν ή και να χρησιμοποιούν πρωτότυπο λογισμικό, που να είναι συμβατό με τα πρότυπα επικοινωνίας οποιουδήποτε άλλου λογισμικού.

- Άρθρο 4.

Θα δημιουργηθεί ένας Κρατικός Οργανισμός, που θα ονομασθεί “Υπηρεσία Τεχνολογιών της Πληροφορίας και των Επικοινωνιών”. Αυτό το Σώμα θα αναφέρεται στον Υπουργό Βιομηχανίας. Η Υπηρεσία ΑΙΤC θα έχει την ευθύνη αναφοράς και παροχής συμβουλευτικών υπηρεσιών προς τις διάφορες Κρατικές υπηρεσίες και τους Δημόσιους οργανισμούς, σχετικά με τη δημιουργία και την αποτύπωση των τεχνικών απαιτήσεων για τις Τεχνολογίες της Πληροφορίας και των Επικοινωνιών. Θα εντοπίζει τις ανάγκες των δημόσιων οργανισμών σε ότι αφορά τον εξοπλισμό και το λογισμικό, θα διασφαλίζει ότι τα πρότυπα επικοινωνίας είναι εναρμονισμένα και θα προτείνει τις τεχνοκρατικές πρακτικές που θα πρέπει να εφαρμόζονται. Θα καταρτίζει απογραφικά μητρώα για κάθε τομέα δραστηριότητας που αφορά τα Ανοικτά Πρότυπα και το διαθέσιμο λογισμικό.

Αναλόγως των αποτελεσμάτων του απογραφικού μητρώου, θα υποστηρίζει την ανάπτυξη των ανοικτών προτύπων και του λογισμικού, που θα κυκλοφορεί μαζί με τον πηγαίο του κώδικα, όπως, επίσης, θα προωθεί τη χρήση αυτού του είδους λογισμικού στον Δημόσιο Τομέα, για να πετύχει τον μετριασμό των οποιωνδήποτε ελλειμμάτων της αγοράς.

Η υπηρεσία ΑΙΤC θα ευνοεί την διαλειτουργικότητα με τα πληροφορικά συστήματα των άλλων χωρών-μελών της Ε.Ε. και θα συμμετέχει σε projects διεθνούς συνεργασίας, στο χώρο των τεχνολογιών της πληροφορίας και των επικοινωνιών.

Η υπηρεσία ΑΙΤC θα διαθέτει από έναν αντιπρόσωπο σε κάθε Νομαρχία.

Ο τρόπος λειτουργίας της υπηρεσίας ΑΙΤC θα καθορίζεται με Υπουργική εντολή.

- Άρθρο 5.

Οι τρόποι εφαρμογής αυτού του Νόμου καθώς και οι όροι για την μετάβαση από την παρούσα κατάσταση, θα καθορισθούν μέσω Διατάγματος που θα εκδώσει το Ανώτατο Κρατικό Συμβούλιο.

- Άρθρο 6.

Τα έξοδα που θα αναλάβει το Κράτος ως αποτέλεσμα της εφαρμογής του Νόμου αυτού, θα πληρωθούν χρησιμοποιώντας τα ποσά που ορίζονται στα άρθρα 575 και 575Α του Γενικού Φορολογικού Κώδικα.

4.3. Νομοσχέδιο που πρότειναν οι Villanueva και Rodrich (Περου)

Θα παρουσιάσουμε τώρα την μετάφραση στα Αγγλικά του αρχικού κειμένου του μεγαλύτερου μέρους από το Νομοσχέδιο υπ. αριθμ. 2485, για τον Νόμο περί Ελεύθερου Λογισμικού στις Δημόσιες Υπηρεσίες, των Περουβιανών Γερουσιαστών Edgar Villanueva Núñez και Jacques Rodrich Ackerman [223].

4.3.1. Συνοπτικό Σκεπτικό

Η συνθετότητα του κόσμου στον οποίο ζούμε, απαιτεί μία αέναη αναθεώρηση και συνεχή αναπροσαρμογή του θεσμικού πλαισίου, για να είναι πάντοτε εναρμονισμένο με τις τρέχουσες τεχνολογικές τάσεις που επιβάλλει η εξέλιξη της ζήτησης.

Η ανακάλυψη νέων Τεχνολογιών της Πληροφορίας και, ανάμεσά τους, το Ελεύθερο Λογισμικό, κατέστη ένα ιδεώδες εργαλείο για την διατήρηση και διαφύλαξη των Κρατικών δεδομένων.

Με τον τρόπο αυτό η τεχνολογία εκπληρώνει τον ρόλο της να διευκολύνει τις διάφορες και πολυποίκιλες ανθρώπινες δραστηριότητες, όντας η διαχείριση της δημόσιας πληροφορίας μία εξ' αυτών.

Σύμφωνα με το Περουβιανό Σύνταγμα, στο Κεφάλαιο 5 του άρθρου 2, "όλοι οι άνθρωποι έχουν το δικαίωμα να ζητούν τις πληροφορίες που τους χρειάζονται, χωρίς να απαιτείται να αποκαλύψουν για ποιο λόγο το πράττουν, και να λαμβάνουν αυτές τις πληροφορίες από οποιαδήποτε Δημόσια Αρχή, εντός των περιόδων που προβλέπονται από τον Νόμο, έναντι ενός εύλογου κόστους. Για τις πληροφορίες που αφορούν ευαίσθητα προσωπικά δεδομένα, ή που αποκλείονται ρητώς από το Νόμο, ή για λόγους εθνικής ασφάλειας, δεν θα υφίσταται η υποχρέωση για την παραχώρησή τους και την αποκάλυψή τους".

Το Κεφάλαιο 6 του ίδιου άρθρου δίνει έμφαση στο δικαίωμα που έχουν όλοι οι άνθρωποι "να τους παρέχεται η εγγύηση ότι οι υπηρεσίες πληροφοριών, είτε σε ψηφιοποιημένη μορφή, είτε όχι, είτε είναι δημόσιες, είτε όχι, δεν θα παραχωρούν πληροφορίες που να αφορούν ευαίσθητα προσωπικά και οικογενειακά δεδομένα ιδιωτικής ζωής".

Με τα παραπάνω δεδομένα, είναι προφανής η ανησυχία του Συντάγματός μας να ορίσει τις θεσμικές βάσεις που θα προστατεύουν την ελευθερία του πολίτη για πρόσβαση στην πληροφόρηση και για την μη-αποκάλυψη των πληροφοριών που αφορούν την προσωπική και την οικογενειακή του ιδιωτική ζωή, όπως ισχύει, επίσης, και για τα θέματα εθνικής ασφαλείας.

Η εγγύηση αυτών των δικαιωμάτων μέσα στο Σύνταγμά μας δεν βασίζεται μόνο στην καλή θέληση των οργάνων του κράτους να τηρούν τους κανόνες του Συντάγματος, αλλά και στη χρήση τεχνολογιών, που σε κάποιες περιπτώσεις συνεισφέρουν, ενώ σε άλλες όχι, προς την κατεύθυνση μίας αποτελεσματικής προστασίας των παραπάνω δικαιωμάτων του πολίτη.

Είναι ακριβώς μέσα σε αυτά τα πλαίσια που για το Κράτος αποκτά ύψιστη σημασία η ενσωμάτωση αυτών των τεχνολογιών, που συνεισφέρουν στην ενίσχυση της πρόσβασης του πολίτη στην πληροφορία, παράλληλα με τις δέουσες επιφυλάξεις για τις περιπτώσεις εκείνες όπου χρειάζεται.

Η χρήση Ελεύθερου Λογισμικού σε όλες τις Κρατικές υπηρεσίες οδηγεί προς αυτή την κατεύθυνση. Ουσιαστικά, μπορούμε να πούμε ότι οι βασικές αρχές που διέπουν το παρόν Νομοσχέδιο σχετίζονται στενά με τις βασικές εγγυήσεις που πρέπει να προσφέρει ένα Δημοκρατικό κράτος και μπορούμε να τις συνοψίσουμε ως εξής:

- 1) Ελεύθερη πρόσβαση των πολιτών προς τις δημόσιες πληροφορίες
- 2) Διατηρησιμότητα στο χρόνο (*perpetuity*) των δημόσιων δεδομένων
- 3) Ασφάλεια του Κράτους και των πολιτών

Για να μπορούν να δωθούν εγγυήσεις για την ελεύθερη πρόσβαση των πολιτών προς τις δημόσιες πληροφορίες, είναι ουσιαστικής σημασίας η κωδικοποίηση των δεδομένων να μην συνδέεται με ένα μόνο προμηθευτή. Η χρήση προδιαγραφών και ανοικτών προτύπων και μορφών αρχείου εγγυάται αυτή την ελεύθερη πρόσβαση, καθιστώντας εφικτή την δημιουργία συμβατού λογισμικού.

Για να μπορούν να δωθούν εγγυήσεις για την διατηρησιμότητα στο χρόνο (*perpetuity*) των δημόσιων δεδομένων, είναι απαραίτητη η χρήση και η συντήρηση του λογισμικού να μην εξαρτώνται από την καλή θέληση των προμηθευτών, ούτε από μονοπωλιακές καταστάσεις που εκείνοι επιβάλλουν. Χρειάζονται συστήματα των οποίων η εξέλιξη να μπορεί να διασφαλισθεί με την διαθεσιμότητα του Πηγαίου Κώδικα.

Για να διασφαλισθεί η εθνική ασφάλεια είναι ζωτικής σημασίας να υπάρχουν συστήματα που δεν περιέχουν στοιχεία που να επιτρέπουν δυνατότητα απομεμακρυσμένου ελέγχου ή αναμετάδοση ανεπιθύμητων πληροφοριών προς τρίτους. Συνεπώς, είναι επιβεβλημένο να έχουμε συστήματα των οποίων ο πηγαίος κώδικας είναι ελεύθερα διαθέσιμος στο ευρύ κοινό, έτσι ώστε ο έλεγχός τους να μπορεί να διεξαχθεί από το Κράτος, από τους πολίτες, καθώς και από έναν μεγάλο αριθμό παγκοσμίως καταξιωμένων και ανεξάρτητων ειδικών.

Αυτή η πρόταση Νόμου προσφέρει περισσότερη ασφάλεια, διότι η γνώση του πηγαίου κώδικα θα εξαλείψει έναν ολοένα αυξανόμενο αριθμό προγραμμάτων με ενσωματωμένα spy-ware.

Κατά τον ίδιο τρόπο, αυτή η πρόταση Νόμου επεκτείνει την ασφάλεια των πολιτών, τόσο υπό την ιδιότητά τους σαν νόμιμοι κάτοχοι των πληροφοριών που διαχειρίζεται το Κράτος, όσο και υπό την ιδιότητά τους σαν καταναλωτές. Σε αυτή την τελευταία περίπτωση, θα δινόταν χώρος για ανάπτυξη μίας εκτεταμένης προσφοράς Ελεύθερου Λογισμικού το οποίο δεν θα διαθέτει εκείνα τα εν δυνάμει spy-ware που καθιστούν δυνατή την καταδυνάστευση της ιδιωτικής ζωής και των ατομικών ελευθεριών.

Το Κράτος, μέσα στη προσπάθειά του να βελτιώσει την ποιότητα της Δημόσιας Διοίκησης, με την ιδιότητά του ως φύλακας και διαχειριστής των ιδιωτικών πληροφοριών, θα θεσμοθετήσει τους όρους κάτω από τους οποίους οι Κρατικές υπηρεσίες θα προμηθεύονται λογισμικό στο μέλλον, δηλαδή, με έναν τρόπο που να είναι συμβατός με τις προβλεπόμενες συνταγματικές εγγυήσεις και τις βασικές αρχές που αναφέρθηκαν προηγουμένως.

Το project προβλέπει ρητώς ότι για να μπορεί ένα οποιοδήποτε λογισμικό να είναι αποδεκτό από το Κράτος, δεν θα πρέπει μόνο να είναι τεχνικώς κατάλληλο να φέρει εις πέρας μία συγκεκριμένη εργασία, αλλά οφείλει να πληροί και ορισμένες προϋποθέσεις σε ότι αφορά θέματα Αδείας, χωρίς την εκπλήρωση των οποίων το Κράτος δεν θα είναι σε θέση να εγγυηθεί προς τους πολίτες την αρμόζουσα επεξεργασία των δεδομένων τους, με ιδιαίτερη έμφαση στην ακεραιότητα των δεδομένων, την εμπιστευτικότητά τους και την επ' αόριστον προσβασιμότητά τους, πράγματα που αποτελούν κρίσιμα στοιχεία για την εκπλήρωση των παραπάνω προϋποθέσεων.

Το Κράτος οριοθετεί τις συνθήκες για την χρήση λογισμικού εκ μέρους των Κρατικών υπηρεσιών, χωρίς να αναμειγνύεται κατά κανέναν τρόπο στις δοσοληψίες του ιδιωτικού τομέως.

Είναι γενικώς αποδεκτό ότι, κατά την σύναψη συμβάσεων, το Κράτος δεν διαθέτει ευρύ φάσμα ελευθεριών σε σύγκριση με τον ιδιωτικό τομέα, καθώς υφίστανται περιορισμοί που απορρέουν από τις απαιτήσεις διαφάνειας σε όλες τις δημόσιες πράξεις και, κατ' αυτή την έννοια, ο κυρίαρχος παράγοντας που θα πρέπει να λαμβάνεται υπ' όψη, όταν καταρτίζεται μία νομοθεσία σε τέτοια θέματα, θα πρέπει να είναι το κοινό συμφέρον.

Το project εγγυάται, επίσης, την αρχή της ισότητας ενώπιον του Νόμου, καθώς δεν αποκλείεται καμία οντότητα (φυσικά ή νομικά πρόσωπα) από το να μπορεί να προμηθευθεί αυτά τα αγαθά, πάντοτε κάτω από τις συνθήκες που αναφέρονται σε αυτό το Νομοσχέδιο και χωρίς κανέναν περαιτέρω περιορισμό, πέραν εκείνων που ορίζονται από τον Νόμο περί σύναψης κρατικών συμβάσεων και προμηθειών (TUO Supreme Decree υπ' αριθμ. 012-2001-PCM).

Πέραν των παραπάνω πλεονεκτημάτων, θα μπορούσαμε να τονίσουμε και τα οφέλη που θα αρχίσουν να αναδύονται σαν συνέπεια των μέτρων αυτών, αμέσως μετά την εφαρμογή τους.

Κατ' αρχάς, δημιουργούνται ευκαιρίας δουλειάς για τους τοπικούς προγραμματιστές. Κατά τη διάρκεια του περασμένου έτους, στον κόσμο των Προγραμμάτων που πωλούνται για διακομιστές στις ΗΠΑ, το 27% ανήκει στην κατηγορία του “Ελεύθερου” λογισμικού, ένα ποσοστό πραγματικά σημαντικό, όταν αναφέρεται σε αυτή την τόσο τεράστια και ανταγωνιστική αγορά. Ο αριθμός αυτός μιλά από μόνος του και αποτελεί μία στιβαρή απάντηση σε εκείνους που νομίζουν πως το

Ελεύθερο Λογισμικό θα έθετε έναν ισχυρό περιορισμό στην απασχόληση των προγραμματιστών ανά την χώρα. Αντιθέτως, η πρωτοβουλία αυτή θα απελευθερώσει μία μεγάλη ποσότητα πόρων και θα προσδώσει ένα κίνητρο για την ενίσχυση της ανθρώπινης δημιουργικότητας.

Μέσω της χρήσης Ελεύθερου Λογισμικού, οι επαγγελματίες μπορούν να αναλύσουν την ρίζα των προβλημάτων και να βελτιώσουν την εξέλιξη σε όποιες περιπτώσεις χρειάζεται, χρησιμοποιώντας Ελεύθερο Λογισμικό, που είναι παγκοσμίως διαθέσιμο υπό διάφορες Άδειες. Πρόκειται για έναν ιδεώδη τομέα, στον οποίο μπορεί να γίνει χρήση της δημιουργικότητας, έναν τομέα στον οποίο οι Περουβιανοί θα μπορούσαν να φθάσουν σε καλά επίπεδα.

Από την άλλη μεριά, με τη βοήθεια του Ελεύθερου Λογισμικού μπορούμε να ξεφορτωθούμε το παράνομο λογισμικό που είναι παρόν σε μερικές Κρατικές υπηρεσίες. Η παράνομη χρήση λογισμικού μέσα στο ίδιο το Κράτος, ή ακόμη και η απλή υπονία ενός τέτοιου γεγονότος, συνιστά ένα ισχυρό κίνητρο για να ωθήσει τον κάθε μέσο δημόσιο υπάλληλο να αλλάξει την κατάσταση που αντιμάχεται τα δικαιώματα πνευματικής ιδιοκτησίας.

Αν και θα ήταν σωστό να πούμε ότι η υιοθέτηση Ελεύθερου Λογισμικού δεν αποτελεί απαραίτητη προϋπόθεση για να σεβαστεί κανείς τον Νόμο, η χρήση του θα μειώσει δραστικά τις ανώμαλες περιπτώσεις και θα λειτουργήσει σαν ένας *τρόπος για την μετάδοση νομιμότητας*, τόσο στα πλαίσια των Κρατικών Υπηρεσιών, όσο και στον ιδιωτικό τομέα.

Μπορούμε να απαριθμήσουμε ήδη πολλές χώρες που ασπάσθηκαν επισήμως την αποκλειστική χρήση Ελεύθερου Λογισμικού στον Δημόσιο τομέα.

Ανάμεσά τους έχουμε την Γαλλία, όπου συζητούν την καθιέρωση μίας νομικής ρύθμισης αυτού του αντικειμένου. Η κυβέρνηση της πόλης του Μεξικού (DF) άρχισε ήδη μία σημαντική μετάβαση προς μία γενικευμένη υιοθέτηση του Ελεύθερου Λογισμικού και αποτελεί την ηγέτιδα χώρα σε αυτό το θέμα στον Δυτικό κόσμο. Στη Βραζιλία, επίσης, η πολιτεία του Recife αποφάσισε την υιοθέτησή του. Η Λαϊκή Δημοκρατία της Κίνας χρησιμοποιεί εδώ και χρόνια Ελεύθερο Λογισμικό, ως συνέπεια μίας κρατικής απόφασης. Το ίδιο ισχύει για τις Σκανδιναβικές χώρες. Στις ΗΠΑ, τόσο η NASA όσο και το Πολεμικό Ναυτικό, μεταξύ άλλων οργανισμών, υιοθέτησαν Ελεύθερο Λογισμικό για κάποιες από τις ανάγκες τους, όπως έπραξαν και πολλές άλλες κυβερνήσεις και ιδιωτικοί φορείς.

Τέλος, το project αναθέτει την εφαρμογή αυτού του Νόμου στην Προεδρία του Υπουργικού Συμβουλίου, ανακηρύσσοντάς την ως υπεύθυνο οργανισμό για τον συντονισμό και την συνέργεια όλων των κυβερνητικών υπηρεσιών. Κατ' αυτή την έννοια, έχει το στρατηγικό πλεονέκτημα να διεξαγάγει αυτή την συγκεκριμένη μεταρρύθμιση και τη διαδικασία μετάβασης από το ιδιοταγές στο Ελεύθερο Λογισμικό.

Αυτό είναι το είδος των ιδεών στα πλαίσια των οποίων διευκρινίσθηκαν τα θέματα αυτά με την παρούσα νομοθετική πρόταση.

4.3.2. Ανάλυση σχέσης Κόστους/ Οφέλους

Αυτή η πρωτοβουλία δεν συνεπάγεται καμία πρόσθετη δαπάνη εκ μέρους του εθνικού κορβανά. Για την εκπλήρωση των σκοπών της, όμως, είναι απαραίτητη η ανακατανομή των κυβερνητικών δαπανών των οποίων η επίπτωση περιορίζεται στα πραγματικά έξοδα του κάθε κυβερνητικού οργανισμού κατά τη διαδικασία κατάρτισης συμβάσεων και κατάθεσης προσφορών προς το Κράτος για την προμήθεια λογισμικού.

Αν και είναι αλήθεια ότι το Ελεύθερο Λογισμικό θα επιφέρει ουσιαστική εξοικονόμηση δαπανών για την κρατική οικονομία, όταν συγκριθεί με το ιδιοταγές λογισμικό, εν τούτοις αυτό δεν συνιστά το κεντρικό επιχείρημα υποστήριξης αυτού του Νομοσχεδίου. Όπως τονίσαμε ήδη, τα πλεονεκτήματά του επικεντρώνονται στις τεχνολογικές εξασφαλίσεις που προσφέρει το πρόγραμμα σε ότι αφορά τις πληροφορίες που διαχειρίζεται το Κράτος, οι οποίες σε πολλές περιπτώσεις είναι εμπιστευτικής φύσεως.

Υπ' αυτή την έννοια, η καλύτερη προστασία των δικαιωμάτων του πολίτη αποτελεί ένα μη μετρήσιμο όφελος που θα πρέπει να λαμβάνεται υπ' όψιν, από την άποψη της ανάλυσης σχέσης κόστους/ οφέλους.

Μπορούμε να συνοψίσουμε τα πλεονεκτήματα του project στις ακόλουθες ομάδες:

- Εθνική Ασφάλεια.

Για να φέρει εις πέρας τις λειτουργίες του, το Κράτος πρέπει να αποθηκεύει και να επεξεργάζεται πληροφορίες για τους πολίτες του. Η σχέση ανάμεσα στο άτομο και το Κράτος εξαρτάται από τον βαθμό ιδιωτικότητας και ακεραιότητας αυτών των δεδομένων, τα οποία πρέπει να προστατεύονται επαρκώς έναντι τριών συγκεκριμένων κινδύνων:

1) Κίνδυνος αποκάλυψης: τα εμπιστευτικά δεδομένα πρέπει να υφίστανται τέτοια διαχείριση ώστε η πρόσβαση σε αυτά γίνεται εφικτή μόνο σε εξουσιοδοτημένα πρόσωπα και σε θεσμούς.

2) Κίνδυνος αδυναμίας πρόσβασης: τα δεδομένα πρέπει να αποθηκεύονται με τέτοιο τρόπο ώστε η πρόσβαση σε αυτά από εξουσιοδοτημένα πρόσωπα και θεσμούς να διασφαλίζεται για όλη την χρονική περίοδο που είναι χρήσιμο.

3) Κίνδυνος τροποποίησης: η τροποποίηση δεδομένων πρέπει να περιορίζεται, και πάλι, μόνο σε εκείνα τα άτομα που είναι εξουσιοδοτημένα για τον σκοπό αυτό.

Με το Ελεύθερο Λογισμικό όλοι αυτοί οι κίνδυνοι μειώνονται σημαντικά. Επιτρέπει στον χρήστη να πραγματοποιεί έναν πλήρη και εξονυχιστικό έλεγχο των μηχανισμών που χρησιμοποιούνται για την διαχείριση των δεδομένων. Το ότι το Ελεύθερο Λογισμικό επιτρέπει τον έλεγχο των πηγών του αποτελεί ένα εξαιρετικό μέτρο ασφάλειας, καθώς το να εκτίθενται οι μηχανισμοί στα μάτια εκπαιδευμένων επαγγελματιών καθιστά την απόκρυψη των κακόβουλων λειτουργιών μέσα στο πρόγραμμα υπερβολικά δυσκολότερο, ακόμη και αν ο τελικός χρήστης δεν διαθέτει τον χρόνο για να ψάχνει μόνος του για κάτι τέτοιο.

- Τεχνολογική Ανεξαρτησία.

Με το ιδιοταγές λογισμικό δεν υπάρχει ισοβαρής ελευθερία ανάμεσα στα συμβαλλόμενα μέρη από την σκοπιά των επεκτάσεων και των διορθώσεων του συστήματος υπό χρήση, και συνεπώς χτίζεται μία τεχνολογική εξάρτηση, στα πλαίσια της οποίας ο προμηθευτής βρίσκεται σε κυρίαρχη θέση, μόνον μίας κατεύθυνσης σε ότι αφορά τους όρους, τις καταληκτικές προθεσμίες και τα κόστη.

Το Ελεύθερο Λογισμικό δίνει στους χρήστες την ελευθερία να ελέγχουν, να διορθώνουν και να τροποποιούν το πρόγραμμα, για να το προσαρμόσουν καλύτερα στις δικές τους ανάγκες. Αυτή η ελευθερία δεν απευθύνεται μόνο προς τους προγραμματιστές. Αν και εκείνοι είναι που θα επωφεληθούν από αυτή σε πρώτο πρόσωπο, αλλά και οι χρήστες θα αντλήσουν μεγάλο όφελος, καθώς με αυτό τον τρόπο μπορούν να προσλάβουν οποιονδήποτε προγραμματιστή (και όχι απαραίτητα τον αρχικό δημιουργό) για να διορθώσει ορισμένα σφάλματα ή για να προσθέσει κάποιες λειτουργίες.

- Τοπική ανάπτυξη.

Στην περίπτωση του ιδιοταγούς λογισμικού, ο χρήστης μπορεί να εκτελέσει ή να τρέξει ένα πρόγραμμα, αλλά δεν είναι σε θέση να το ελέγξει ή να το τροποποιήσει. Κατά συνέπεια, ο χρήστης δεν μπορεί να μάθει τίποτε από αυτό. Οι χρήστες γίνονται εξαρτημένοι από μία τεχνολογία που όχι μόνον δεν κατανοούν, αλλά η πρόσβαση στην οποία τους απαγορεύεται ρητώς. Οι επαγγελματίες του χώρου, που θα μπορούσαν να βοηθήσουν τους χρήστες να πετύχουν τους σκοπούς τους, είναι εξ' ίσου περιορισμένοι: αφού ο τρόπος με τον οποίο δουλεύει το πρόγραμμα είναι μυστικός και ο έλεγχός του δεν επιτρέπεται, δεν είναι δυνατόν να διορθωθεί. Με αυτό τον τρόπο, οι εντόπιοι επαγγελματίες βλέπουν τις δικές τους πιθανότητες να μπορέσουν να προσφέρουν προστιθέμενη αξία, να μειώνονται ολοένα και περισσότερο καθώς και τους ορίζοντες της απασχόλησής τους να στενεύουν, παράλληλα με τις δυνατότητές τους να μάθουν παραπάνω πράγματα. Με το Ελεύθερο Λογισμικό, αυτά τα μειονεκτήματα του ιδιοταγούς λογισμικού μετριάζονται σε μεγάλο βαθμό.

- Κόστος του λογισμικού.

Το κόστος μειώνεται πάρα πολύ διότι, αφού είναι ελεύθερο, δεν υπάρχει ανάγκη προς αναζήτηση πρόσθετων Αδειών για να συνεχίσει η χρήση του προγράμματος. Αυτή η ανάγκη υφίσταται με το ιδιοταγές λογισμικό. Είναι σημαντικό να μπορεί ο χρήστης να κρατήσει αυτά τα κόστη υπό έλεγχο, διότι αν δεν το κατορθώσει, δεν θα είναι σε θέση να συνεχίσει περαιτέρω με τους σκοπούς του, καθώς θα δεσμεύεται από απρόβλεπτες περιστάσεις. Ορίστε, και πάλι, η τεχνολογική εξάρτηση που απειλεί το Ελεύθερο Λογισμικό.

- Περισσότερες πηγές απασχόλησης.

Με το Ελεύθερο Λογισμικό, αποδεσμεύονται ανθρώπινες εργατοώρες και ανθρώπινο δυναμικό που δεσμευόταν μέσω της τεχνολογικής εξάρτησης του Κράτους από το ιδιοταγές λογισμικό. Τώρα θα μπορούν να αποδωθούν εκ νέου πόροι χρηστών (σε αυτή την περίπτωση, οι Κρατικές υπηρεσίες) για τη συντήρηση και υποστήριξη του Ελεύθερου Λογισμικού.

- Ενίσχυση της δημιουργικότητας και της επιχειρηματικότητας.

4.3.2.1. Κόστη

Το μεγάλο κόστος που συνεπάγεται η μετάβαση από το ιδιοταγές στο Ελεύθερο Λογισμικό περιορίζεται μόνο σε αυτή τη διαδικασία μετατροπής. Ακόμη κι αν είναι αλήθεια ότι η διαδικασία μετάβασης επιφέρει κόστη σε μελέτες, λήψη αποφάσεων για την υλοποίηση των νέων συστημάτων, εργατοώρες για την υλοποίηση των αλλαγών, την μετατροπή των δεδομένων, την επανεκπαίδευση του προσωπικού και ενδεχομένως έξοδα σε Άδειες ή/και σε ανάπτυξη και χρόνο, είναι εξ' ίσου βέβαιο πως όλα αυτά είναι εφ' άπαξ κόστη που θα καταβληθούν μόνον μία φορά.

Από την άλλη μεριά, και το ιδιοταγές λογισμικό έχει τα δικά του έξοδα, τα οποία άπαξ και καταβληθούν, δεν γίνεται να ανακτηθούν. Πέρα, όμως, από αυτά τα κόστη, υπάρχουν και άλλα που απορρέουν από το ιδιοταγές λογισμικό: οι συνεχείς ενημερώσεις (που ενίοτε επιβάλλονται από ένα αυτοσυντηρούμενο μονοπώλιο) και, πάνω από όλα, το τεράστιο τίμημα που αποτελεί για το Κράτος η απώλεια των ελευθεριών εκείνων που εγγυώνται τον έλεγχο των ίδιων του των πληροφοριών.

Αυτά αντιπροσωπεύουν πάγια και συνεχή κόστη και με το πέρασμα του χρόνου, αργά ή γρήγορα, ξεπερνούν τα κόστη της εφαρμογής μίας μετάβασης.

Για να συνοψίσουμε, τα πλεονεκτήματα της διαδικασίας μετάβασης υπερτερούν έναντι του σχετικού κόστους.

4.3.3. Νομική φόρμουλα

4.3.3.1. Άρθρο 1. Σκοπός του Νόμου

Να χρησιμοποιείται αποκλειστικά Ελεύθερο Λογισμικό σε όλα τα συστήματα και τον υπολογιστικό εξοπλισμό όλων των Κρατικών υπηρεσιών.

4.3.3.2. Άρθρο 2. Σκοπός της εφαρμογής

Η Εκτελεστική, η Νομοθετική και η Δικαστική εξουσία, καθώς και οι Περιφερειακοί και Τοπικοί αποκεντρωμένοι οργανισμοί αυτοδιοίκησης, μαζί με τις υπηρεσίες του ευρύτερου δημόσιου τομέα όπου το Κράτος διαθέτει την πλειοψηφία των μετοχών, θα χρησιμοποιούν Ελεύθερο Λογισμικό στα συστήματά τους και τον υπολογιστικό τους εξοπλισμό.

4.3.3.3. Άρθρο 3. Υπεύθυνη Αρχή για την εφαρμογή

Η Αρχή που θα έχει την ευθύνη για την εφαρμογή του Νόμου θα είναι το Υπουργικό Συμβούλιο.

4.3.3.4. Άρθρο 4. Ορισμός του Ελεύθερου Λογισμικού

Για τους σκοπούς αυτού του Νόμου, ως *Ελεύθερο πρόγραμμα* ή *Ελεύθερο Λογισμικό* ορίζεται εκείνο του οποίου η Άδεια θα εγγυάται προς τον χρήστη, χωρίς κανένα πρόσθετο κόστος, τα ακόλουθα:

- Απεριόριστη χρήση του προγράμματος για οποιαδήποτε χρήση.
- Απεριόριστη πρόσβαση στον αντίστοιχο πηγαίο κώδικα.
- Δυνατότητα εξονυχιστικού ελέγχου του μηχανισμού λειτουργίας του προγράμματος.
- Χρήση των εσωτερικών μηχανισμών **και οποιουδήποτε τμήματος του λογισμικού που έχει επιλεχθεί αυθαίρετα**, για να προσαρμοσθούν στις ανάγκες του χρήστη.
- Ελευθερία δημιουργίας και διανομής αντιγράφων του λογισμικού.
- Τροποποίηση του λογισμικού και ελευθερία αναδιανομής του προκύπτοντος νέου λογισμικού με τις προαναφερθείσες τροποποιήσεις, υπό την ίδια Άδεια του αρχικού λογισμικού.

4.3.3.5. Άρθρο 5. Εξαιρέσεις

Στην περίπτωση όπου δεν υπάρχει μία λύση που να χρησιμοποιεί Ελεύθερο Λογισμικό, και η οποία να ικανοποιεί τις επιθυμητές ανάγκες, τότε οι Κρατικές Υπηρεσίες θα μπορούσαν να υιοθετήσουν τις ακόλουθες εναλλακτικές, κατά σειρά προτεραιότητας.

Εάν, κατά την προσπάθεια επίλυσης ενός τεχνικού προβλήματος, σημειωθούν τεκμηριώσιμοι χρονικοί περιορισμοί και εντοπισθεί η διαθεσιμότητα ιδιοταγούς λογισμικού, τότε ο Οργανισμός που το χρειάζεται μπορεί να διαπραγματευθεί με την εποπτεύουσα αρχή μία άδεια εξαίρεσης, για να χρησιμοποιήσει ένα ιδιοταγές λογισμικό που να διαθέτει τα εξής χαρακτηριστικά:

- Τα προγράμματα αυτά θα συμμορφώνονται με τους όρους που αναφέρονται στο Κεφάλαιο 4 του Νόμου, εκτός από την ελεύθερη διανομή του τροποποιημένου προγράμματος. Σε αυτή την περίπτωση, η άδεια εξαίρεσης θα μπορούσε να καταστεί οριστική.
- Αν δεν υπάρχουν προγράμματα που να ανήκουν στην προηγούμενη κατηγορία, τότε θα επιλεγούν εκείνα που υπάρχουν στα πλαίσια ενός Ελεύθερου project σε προχωρημένη φάση ανάπτυξης. Η άδεια, σε αυτή την περίπτωση, θα είναι παροδική και θα λήγει αυτομάτως όταν το Ελεύθερο Λογισμικό ωριμάσει με το είδος λειτουργιών που είναι απαραίτητες.

• Αν δεν βρεθούν προϊόντα που να πληρούν αυτές τις προϋποθέσεις, τότε μπορεί να χρησιμοποιηθεί ιδιοταγές λογισμικό, αλλά η άδεια εξαίρεσης που θα έχει ζητηθεί από την αρμόδια αρχή θα λήξει αυτόματα 2 χρόνια μετά την έκδοσή της. Η ενδεχόμενη ανανέωσή της θα απαιτεί πρώτα τη βεβαίωση ότι δεν ήταν διαθέσιμη μία ικανοποιητική αντίστοιχη λύση σε Ελεύθερο Λογισμικό. Η αρμόδια Αρχή θα εκδώσει μία άδεια εξαίρεσης μόνον αν ο Κρατικός Οργανισμός εγγυάται την αποθήκευση των δεδομένων με Ανοικτά Πρότυπα, χωρίς προκατάληψη πληρωμής για τις ιδιοταγείς Άδειες.

4.3.3.6. Άρθρο 6. Εκπαιδευτικές Άδειες

Όλα τα Εκπαιδευτικά Ιδρύματα που εξαρτώνται από το Κράτος έχουν την δυνατότητα να διαχειρίζονται την ιδιοταγή άδεια για την χρήση της στην έρευνα, κατόπιν καταβολής των αντίστοιχων δικαιωμάτων πνευματικής ιδιοκτησίας και τις σχετικές άδειες, υπό την προϋπόθεση ότι ο σκοπός της έρευνας σχετίζεται άμεσα με την χρήση του εν λόγω προγράμματος.

4.3.3.7. Άρθρο 7. Διαφάνεια των εξαιρέσεων

Οι εξαιρέσεις που παραχωρούνται αναφορικά με μία δεδομένη εφαρμογή θα πρέπει να είναι τεκμηριωμένες και να δημοσιεύονται στην ιστοσελίδα του Κρατικού Portal.

Η απόφαση που θεσμοθετεί την εξαίρεση θα πρέπει να απαριθμεί τις λειτουργικές απαιτήσεις που πρέπει να πληροί το πρόγραμμα.

4.3.3.8. Άρθρο 8. Εξαιρετική εξουσιοδότηση

Σε περίπτωση που κάποια κρατική υπηρεσία δεν μπορεί να εκπληρώσει τις απαιτήσεις για το λογισμικό που αναφέρονται στο άρθρο 2 αυτού του Νόμου, τότε εξουσιοδοτείται να προβεί σε αγορά ιδιοταγούς λογισμικού για να επεξεργασθεί ή να αποθηκεύσει δεδομένα τα οποία θα πρέπει να τηρήσει υπό την φύλαξή της, ενώ η αντίστοιχη Αρχή υποχρεούται να δημοσιεύσει στην Κρατική πύλη (portal) μία αναφορά που θα εξηγεί τους κινδύνους που συνδέονται με τη χρήση του συγκεκριμένου λογισμικού για να επιτελεσθεί μία δουλειά.

Οι εξαιρετικές άδειες που παραχωρούνται προς Κρατικές υπηρεσίες και που σχετίζονται με την ασφάλεια και την εθνική άμυνα, απαλλάσσονται από την προαναφερθείσα υποχρέωση.

4.3.3.9. Άρθρο 9. Κατανομή ευθυνών

Η ανώτατη διοικητική Αρχή καθώς και η Τεχνική και Πληροφορική αρχή της κάθε Κρατικής υπηρεσίας, αναλαμβάνουν την ευθύνη για την πλήρη εφαρμογή του Νόμου αυτού.

4.3.3.10. Άρθρο 10. Κανονιστικός νόμος

Ο εκτελεστικός κλάδος της κυβέρνησης θα ορίζει, εντός προθεσμίας 180 ημερών, τους όρους, τις καταληκτικές ημερομηνίες και τη μορφή υπό την οποία το τρέχον status quo θα αλλάξει σε ένα άλλο, που θα καλύπτει τις απαιτήσεις του νόμου και θα διέπει, κατ' αυτή την έννοια, όλες τις μελλοντικές συμβάσεις και τις διαπραγματεύσεις για προμήθειες λογισμικού.

Παρομοίως, θα κατευθύνει τη διαδικασία μετάβασης από τα ιδιοταγή συστήματα λογισμικού προς τα αντίστοιχα συστήματα ελεύθερου λογισμικού, σε όλες τις περιπτώσεις όπου οι δεδομένες περιστάσεις το απαιτούν.

4.3.3.11. Άρθρο 11. Γλωσσάρι των όρων

- α) Πρόγραμμα ή Λογισμικό: οποιαδήποτε σειρά οδηγιών που χρησιμοποιείται από ένα σύστημα επεξεργασίας ψηφιακών δεδομένων, για να φέρει εις πέρας ένα συγκεκριμένο έργο ή να επιλύσει ένα πρόβλημα.
- β) Εκτέλεση ή χρήση ενός προγράμματος: η πράξη της χρησιμοποίησής του σε οποιοδήποτε σύστημα επεξεργασίας ψηφιακών δεδομένων για την διεξαγωγή μίας λειτουργίας.
- γ) Χρήστης: φυσικό ή νομικό πρόσωπο που χρησιμοποιεί το λογισμικό.
- δ) Πηγαίος κώδικας ή Πηγαίο πρόγραμμα: ένα πλήρες σύνολο οδηγιών και πηγαίων ψηφιακών αρχείων, που δημιουργήθηκαν ή τροποποιήθηκαν από εκείνους που τα τα προγραμμάτισαν, συν όλα εκείνα τα ψηφιακά αρχεία υποστήριξης, όπως πίνακες δεδομένων (data tables), εικόνες, τεχνικές προδιαγραφές, έγγραφη τεκμηρίωση, και οποιοδήποτε άλλο στοιχείο που είναι απαραίτητο για να φτιαχτεί το εκτελέσιμο αρχείο.
- Σαν μία εξαίρεση, όλα εκείνα τα εργαλεία που είναι συνήθως διαθέσιμα υπό μορφή Ελεύθερου Λογισμικού σε άλλα μέσα, μπορούν να εξαιρεθούν, όπως για παράδειγμα, οι μεταγλωττιστές, τα λειτουργικά συστήματα και οι βιβλιοθήκες.
- ε) Ελεύθερο Λογισμικό ή Πρόγραμμα: εκείνο που εγγυάται στον χρήστη, χωρίς κανένα πρόσθετο κόστος, τα ακόλουθα:
- Απεριόριστη χρήση του προγράμματος για οποιονδήποτε σκοπό.
 - Απεριόριστη πρόσβαση στον αντίστοιχο πηγαίο κώδικα.
 - Εξονυχιστικό έλεγχο των μηχανισμών λειτουργίας του προγράμματος.
 - Χρήση των εσωτερικών μηχανισμών και αυθαίρετων τμημάτων του λογισμικού, για την προσαρμογή τους στις ανάγκες του χρήστη.
 - Ελευθερία να φτιάχνει και να διανέμει αντίτυπα του λογισμικού.
 - Τροποποίηση του λογισμικού και ελευθερία αναδιανομής των παραπάνω τροποποιήσεων του νέου προγράμματος που προέκυψε, υπό τους όρους της ίδιας Άδειας του αρχικού λογισμικού.
- στ) Ιδιοταγές λογισμικό (μη-ελεύθερο λογισμικό): εκείνο που δεν πληροί όλες τις προϋποθέσεις που αναφέρονται στην προηγούμενη παράγραφο.
- ζ) Ανοικτή μορφή αρχείου: οποιαδήποτε μορφή ψηφιακά κωδικοποιημένης πληροφορίας που πληροί συγχρόνως τις υπάρχουσες προδιαγραφές και τις ακόλουθες προϋποθέσεις:
- Η έγγραφη τεχνική τεκμηρίωση είναι δημοσίως διαθέσιμη.
 - Ο πηγαίος κώδικας τουλάχιστον μίας πλήρους υλοποίησης αναφοράς (reference implementation) είναι δημοσίως διαθέσιμος.
 - Δεν υπάρχουν περιορισμοί για την δημιουργία προγραμμάτων που να αποθηκεύουν, να αναμεταδίδουν, να λαμβάνουν ή να προσπελάζουν δεδομένα που κωδικοποιήθηκαν με τέτοιο τρόπο.

4.4. Επιστολές από την Microsoft Περού και τον Γερουσιαστή Villanueva

Στις 21 Μαρτίου 2002, ο Juan Alberto González, ο Γενικός Διευθυντής της Microsoft Peru, έστειλε μία επιστολή προς τον Γερουσιαστή Edgar Villanueva Núñez, σχετικά με το Νομοσχέδιο που είχε

καταθέσει για το Ελεύθερο Λογισμικό [129]. Στις 8 Απριλίου, ο Γερουσιαστής απάντησε [179]. Παραθέτουμε εδώ την Αγγλική μετάφραση της κυριολεκτικής καταγραφής σχεδόν ολοκλήρου του κειμένου και των δύο επιστολών (παραλείφθηκαν οι παράγραφοι που δεν σχετίζονται με το Νομοσχέδιο).

4.4.1. Επιστολή από την Microsoft Peru

Σύμφωνα με όσα κανονίσαμε στη συνάντησή μας, παρακολουθήσαμε το forum που οργανώθηκε στη Γερουσία της Δημοκρατίας, στις 6 Μαρτίου, για το σχέδιο Νόμου που καταθέσατε. Εκεί είχαμε την ευκαιρία να ακούσουμε διάφορες παρουσιάσεις επί του θέματος. Θα θέλαμε τώρα να παρουσιάσουμε και τη δική μας θέση, ούτως ώστε να έχετε μία πληρέστερη εικόνα για την πραγματική κατάσταση.

Η πρότασή σας επιβάλλει η κάθε Δημόσια υπηρεσία να χρησιμοποιεί κατ' αποκλειστικότητα Ελεύθερο Λογισμικό, που είναι επίσης γνωστό και σαν *Λογισμικό Ανοικτού Κώδικα*. Αυτό είναι κάτι που παραβιάζει τις αρχές της ισότητας ενώπιον του Νόμου, της μη διάκρισης, της ελεύθερης ιδιωτικής πρωτοβουλίας, και τις ελευθερίες της βιομηχανίας και της σύναψης συμβάσεων (*industry and contracting*), οι οποίες προστατεύονται από το Σύνταγμα.

Η πρότασή σας, επιβάλλοντας τη χρήση *Λογισμικού Ανοικτού Κώδικα*, ξεκινά μία μεταχείριση μη-ανταγωνιστική και με διακρίσεις, στη σύναψη συμβάσεων και προμηθειών από τους Δημόσιους οργανισμούς, παραβιάζοντας τις βασικές αρχές του Νόμου του Κράτους περί Σύναψης Δημόσιων Συμβάσεων και Προμηθειών (Αριθμ. 26850).

Αναγκάζοντας το Κράτος να προτιμήσει ένα επιχειρηματικό μοντέλο με αποκλειστική υποστήριξη του *Λογισμικού Ανοικτού Κώδικα*, η πρότασή σας θα αποθαρρύνει τους εντόπιους αλλά και τους διεθνείς κατασκευαστές λογισμικού οι οποίοι προωθούν πραγματικές και σημαντικές επενδύσεις στη χώρα, δημιουργούν ένα σημαντικό αριθμό άμεσων και έμμεσων θέσεων εργασίας, και συνεπώς συνεισφέρουν στα εθνικά έσοδα. Αντιθέτως, οι τρόποι ανάπτυξης του *Λογισμικού Ανοικτού Κώδικα* αποφέρουν μικρότερα οφέλη για την οικονομία, αφού δημιουργούν κυρίως θέσεις στον τομέα των υπηρεσιών.

Η πρότασή σας επιβάλλει τη χρήση Ανοικτού Λογισμικού χωρίς να λαμβάνει υπ' όψη τους κινδύνους που αυτό συνεπάγεται για την ασφάλεια, την εξασφάλιση, ακόμη και την πιθανή παραβίαση των δικαιωμάτων πνευματικής ιδιοκτησίας τρίτων.

Προϋποθέτει, εσφαλμένα, ότι το *Λογισμικό Ανοικτού Κώδικα* είναι Δωρεάν Λογισμικό, δηλαδή χωρίς κόστος, φθάνοντας έτσι σε λάθος συμπεράσματα για το πόσα χρήματα θα εξοικονομήσει το Κράτος. Δεν προσφέρει μία ανάλυση κόστους/ οφέλους, για να υποστηρίξει αυτή την υπόθεση.

Είναι εσφαλμένη η σκέψη ότι το *Λογισμικό Ανοικτού Κώδικα* είναι δωρεάν. Μία έρευνα που έκαναν οι Gartner Group (σημαντικοί και διεθνώς γνωστοί ερευνητές της αγοράς σε θέματα του κόσμου της τεχνολογίας), έδειξε ότι το κόστος κτήσης του λογισμικού (λειτουργικό σύστημα και εφαρμογές) είναι μόνο το 8% του συνολικού κόστους ιδιοκτησίας που οι επιχειρήσεις και οι οργανισμοί πρέπει να αντιμετωπίσουν ως συνέπεια της ορθολογικής και παραγωγικής χρήσης της τεχνολογίας. Το υπόλοιπο 92% αντιπροσωπεύει κόστη εγκατάστασης, εκπαίδευσης, υποστήριξης, συντήρησης, διαχείρισης και downtime.

Ένα από τα επιχειρήματα που στηρίζουν την πρότασή σας είναι το υποτιθέμεν φθηνότερο τίμημα

του λογισμικού ανοικτού κώδικα, σε σύγκριση με το εμπορικό λογισμικό, χωρίς να λαμβάνεται υπ' όψη η δυνατότητα των μοντέλων μαζικής αδειοδότησης (*volume licensing*). Το Κράτος μπορεί πραγματικά να επωφεληθεί από αυτά, όπως και σε άλλες χώρες.

Επιπροσθέτως, η προσέγγιση που επιλέχθηκε από το δικό σας project (i) είναι σαφώς δαπανηρότερη, λόγω του υψηλού κόστους της μετάβασης, (ii) διακινδυνεύει την απώλεια διαλειτουργικότητας μεταξύ πληροφοριακών συστημάτων, τόσο εντός της Κρατικής μηχανής, όσο και ανάμεσα στο Κράτος και τον ιδιωτικό τομέα, λόγω των πολλών διαφορετικών διανομών του λογισμικού ανοικτού κώδικα που κυκλοφορούν στην αγορά.

Στις περισσότερες περιπτώσεις, το λογισμικό ανοικτού κώδικα δεν προσφέρει ικανοποιητικά επίπεδα υπηρεσιών για να μπορέσουν οι χρήστες του να πετύχουν μία αυξημένη παραγωγικότητα, ούτε προσφέρει εγγυήσεις από γνωστούς κατασκευαστές. Αυτά τα πράγματα οδήγησαν πολλούς δημόσιους οργανισμούς στο να ανακαλέσουν τις αποφάσεις τους περί χρήσης λογισμικού ανοικτού κώδικα. Τώρα, στη θέση του, χρησιμοποιούν εμπορικό λογισμικό.

Αυτό το project αποθαρρύνει τη δημιουργικότητα στην Περουβιανή βιομηχανία λογισμικού, που πωλεί αγαθά αξίας 40 εκατομμυρίων δολαρίων ετησίως, εκ των οποίων 4 εκατομμύρια δολάρια αντιστοιχούν σε εξαγωγές (10^η θέση στη κατάταξη των μη-παραδοσιακών εξαγωγών του Περού, παραπάνω από τα χειροποίητα αγαθά) και αποτελεί μία πηγή για θέσεις εργασίας υψηλής εξειδίκευσης. Με έναν νόμο που ενθαρρύνει τη χρήση λογισμικού ανοικτού κώδικα, οι προγραμματιστές χάνουν τα παραδοσιακά τους δικαιώματα πνευματικής ιδιοκτησίας και την πλέον σημαντική τους πηγή κερδών.

Από τη στιγμή που το λογισμικό ανοικτού κώδικα μπορεί να διανέμεται ελεύθερα, δεν μπορεί να οδηγήσει καθόλου στη δημιουργία κερδών για τους προγραμματιστές του μέσω των εξαγωγών. Με αυτό τον τρόπο, εξασθενίζει το πολλαπλασιαστικό αποτέλεσμα των πωλήσεων λογισμικού προς άλλες χώρες και ανακόπτει την πορεία ανάπτυξης της τοπικής βιομηχανίας, την οποία το Κράτος θα έπρεπε να υποβοηθά.

Στο forum, συζητήθηκε, επίσης, η σημασία της χρήσης του λογισμικού ανοικτού κώδικα στην Εκπαίδευση, χωρίς να γίνεται κανένα σχόλιο για την πλήρη αποτυχία αυτής της πρωτοβουλίας σε χώρες σαν το Μεξικό. Εκεί, οι ίδιοι οι αξιωματούχοι του Κράτους που υποστήριζαν το project, τώρα λένε ότι το λογισμικό ανοικτού κώδικα δεν προσέφερε μία εκπαιδευτική εμπειρία προς τα παιδιά των Σχολείων, δεν υπήρχαν κατάλληλα επίπεδα εκπαίδευσης σε πανεθνικό επίπεδο, παρέχονταν ακατάλληλη υποστήριξη προς την πλατφόρμα και το λογισμικό δεν ήταν επαρκώς ενσωματωμένο με τα υπάρχοντα υπολογιστικά Σχολικά συστήματα.

Αν το λογισμικό ανοικτού κώδικα πληρούσε όλες τις προϋποθέσεις που απαιτούνται από τις κρατικές υπηρεσίες, τότε για ποιο λόγο χρειαζόταν να θεσπισθεί ένας νόμος για τη χρήση του; Δεν θα έπρεπε η αγορά να μπορεί να επιλέγει ελεύθερα ποια προϊόντα παρέχουν τα περισσότερα πλεονεκτήματα και τη μεγαλύτερη αξία;

4.4.2. Απάντηση από τον Γερουσιαστή Villanueva

Πρώτα απ' όλα, σας ευχαριστώ για την επιστολή σας με ημερομηνία 25 Μαρτίου 2002, στην οποία εκθέτετε την επίσημη θέση της Microsoft αναφορικά με το Νομοσχέδιο υπ' αριθμόν 1609, περί του Ελεύθερου Λογισμικού στη Δημόσια Διοίκηση, που εμπνέεται αναμφισβήτητα από την επιθυμία του Περού να βρει μία κατάλληλη θέση στο συνολικό τεχνολογικό πλαίσιο. Στο ίδιο πνεύμα, και όντας πεπεισμένοι ότι θα βρούμε τις καλύτερες λύσεις μέσω της ανταλλαγής καθαρών και ανοιχτών

ιδεών, θα εκμεταλλευθώ αυτή την ευκαιρία για να απαντήσω στους σχολιασμούς που περιλαμβάνετε στην επιστολή σας.

Παρ' όλο που αναγνωρίζω ότι γνώμες σαν την δική σας αποτελούν μία σημαντική συνεισφορά, θα ήταν, όμως, ακόμη πιο χρήσιμο για εμάς αν, αντί να αντιτάσσετε αντιρρήσεις γενικής φύσεως (τις οποίες θα αναλύσουμε λεπτομερώς πιο κάτω), είχατε παραθέσει στιβαρά επιχειρήματα για τα πλεονεκτήματα που το ιδιοταγές λογισμικό θα μπορούσε να αποφέρει για το Περουβιανό Κράτος και τους πολίτες του, γενικότερα, αφού αυτό θα μας επέτρεπε μία πιο πεφωτισμένη ανταλλαγή και εξέταση των αμοιβαίων μας απόψεων.

Αφού έχουμε την πρόθεση να κάνουμε μία πολιτισμένη συζήτηση αντιπαράθεσης (*debate*), θα θεωρήσουμε ότι εκείνο που εσείς αποκαλείτε *λογισμικό ανοικτού κώδικα*, αντιστοιχεί σε εκείνο που το Νομοσχέδιο ορίζει ως *Ελεύθερο Λογισμικό*, αφού υπάρχει και λογισμικό που διανέμεται μαζί με τον πηγαίο του κώδικα, αλλά χωρίς να εμπίπτει στον ορισμό που υιοθετεί το Νομοσχέδιο. Επιπλέον, προϋποθέτουμε ότι εκείνο που εσείς αποκαλείτε *εμπορικό λογισμικό*, είναι το ίδιο με εκείνο που το σχέδιο νόμου ορίζει ως *ιδιοταγές ή μη-ελεύθερο*, αφού υπάρχει ελεύθερο λογισμικό που διατίθεται προς πώληση στην αγορά έναντι ενός τιμήματος, ακριβώς σαν οποιοδήποτε άλλο αγαθό ή υπηρεσία.

Είναι, επίσης, απαραίτητο να διευκρινισθεί ότι ο σκοπός του νομοσχεδίου που συζητάμε δεν σχετίζεται άμεσα με το συνολικό ποσό άμεσης εξοικονόμησης που θα μπορούσε να επιτευχθεί με τη χρήση Ελεύθερου Λογισμικού στους Κρατικούς Οργανισμούς. Αυτό το ποσό, σε κάθε περίπτωση, δεν είναι παρά μία επιπλέον προστιθέμενη αξία, αλλά σε καμία περίπτωση δεν αποτελεί την κυρίως εστίαση του νομοσχεδίου. Οι βασικές αρχές που διαπνέουν το νομοσχέδιο συνδέονται με τις βασικές εγγυήσεις ενός Κράτους δικαίου, όπως:

- η ελεύθερη πρόσβαση του πολίτη στις δημόσιες πληροφορίες,
- η εγγύηση της μακροχρόνιας διατήρησης και διαφύλαξης των δημόσιων δεδομένων,
- η ασφάλεια του Κράτους και των πολιτών.

Για να εξασφαλισθεί η ελεύθερη πρόσβαση των πολιτών προς τις δημόσιες πληροφορίες, είναι βασικής σημασίας η μορφή κωδικοποίησης των δεδομένων να μην συνδέεται με έναν μόνο προμηθευτή. Η χρήση Ανοιχτών προδιαγραφών και Προτύπων προσφέρει μία εγγύηση γι' αυτή την ελεύθερη πρόσβαση, και εάν αυτό απαιτηθεί, μέσω της δημιουργίας συμβατού λογισμικού.

Για να εξασφαλισθεί η εγγύηση της μακροχρόνιας διατήρησης και διαφύλαξης των δημόσιων δεδομένων, είναι απαραίτητο η χρηστικότητα και η συντήρηση του λογισμικού να μην εξαρτώνται μόνο από την καλή θέληση των προμηθευτών, ή από τους μονοπωλιακούς όρους που εκείνοι επιβάλλουν. Για τους λόγους αυτούς, το Κράτος χρειάζεται συστήματα των οποίων η ανάπτυξη να μπορεί να διασφαλίζεται χάρη στην διαθεσιμότητα του πηγαίου κώδικα.

Για να εξασφαλισθεί η Εθνική ασφάλεια ή η Κρατική ασφάλεια, είναι επιβεβλημένο να μπορούμε να βασιστούμε σε συστήματα που δεν επιτρέπουν τον έλεγχο εξ' αποστάσεως ή την ανεπιθύμητη μετάδοση πληροφοριών προς τρίτους. Συνεπώς, χρειάζονται συστήματα με πηγαίο κώδικα πλήρως προσβάσιμο στο κοινό, για να μπορούν έτσι να ελέγχονται τόσο από το ίδιο το Κράτος, αλλά και από τους πολίτες, καθώς και από πολλούς ανεξάρτητους ειδικούς ανά τον κόσμο. Η πρότασή μας προσφέρει παραπάνω ασφάλεια, αφού η γνώση του πηγαίου κώδικα θα εξαφανίσει τον αυξανόμενο αριθμό των προγραμμάτων που έχουν μέσα τους κατασκοπευτικό κώδικα (*spy code*).

Με τον ίδιο τρόπο, η πρότασή μας ενισχύει την ασφάλεια των πολιτών, τόσο υπό την ιδιότητά τους ως νομίμων κατόχων των πληροφοριών που διαχειρίζεται το Κράτος, όσο και υπό την ιδιότητά τους ως καταναλωτών, επιτρέποντας, σε αυτή τη δεύτερη περίπτωση, την ανάπτυξη μίας μεγάλης ποικιλίας Ελεύθερου Λογισμικού, που δεν θα ενσωματώνει κατασκοπευτικό κώδικα (*spy code*), που θα έθετε υπό κίνδυνο την ιδιωτικότητα και τις ατομικές ελευθερίες.

Κατ' αυτή την έννοια, το Νομοσχέδιο περιορίζεται να ορίσει τους όρους υπό τους οποίους οι Κρατικοί Οργανισμοί θα προμηθεύονται λογισμικό στο μέλλον, δηλαδή, κατά έναν τρόπο που να είναι συμβατός με αυτές τις βασικές αρχές.

Κατά την ανάγνωση του Νομοσχεδίου θα καταστεί προφανές ότι, άπαξ και ψηφισθεί:

- ο Νόμος δεν απαγορεύει την παραγωγή ιδιοταγούς λογισμικού
- ο Νόμος δεν απαγορεύει την πώληση ιδιοταγούς λογισμικού
- ο Νόμος δεν διευκρινίζει ποια συγκεκριμένα λογισμικά να χρησιμοποιούνται
- ο Νόμος δεν επιβάλλει τον προμηθευτή από τον οποίο πρέπει να αγοράζεται το λογισμικό
- ο Νόμος δεν περιορίζει τους όρους υπό τους οποίους μπορεί να αδειοδοτείται ένα προϊόν.

Εκείνο που το Νομοσχέδιο εκφράζει σαφώς, είναι ότι, για να είναι αποδεκτό το λογισμικό από την πλευρά του Κράτους, δεν είναι επαρκές να είναι τεχνικώς ικανό να εκτελεί μία λειτουργία, αλλά, επιπλέον, θα πρέπει οι όροι των συμβάσεων να πληρούν μία σειρά προϋποθέσεων σε ότι αφορά την Άδεια, χωρίς τις οποίες το Κράτος δεν μπορεί να εγγυηθεί προς τους πολίτες την αρμόζουσα επεξεργασία των δεδομένων, την επίβλεψη της ακεραιότητάς τους, την εμπιστευτικότητα και την προσβασιμότητά τους σε βάθος χρόνου, καθώς όλα αυτά αποτελούν θέματα κρίσιμης σημασίας για τη φυσιολογική του λειτουργία.

Συμφωνούμε, κε. González, ότι οι τεχνολογίες της πληροφορίας και των επικοινωνιών έχουν μία επίπτωση πολύ σημαντική πάνω στην ποιότητα ζωής των πολιτών (είτε θετική, είτε αρνητική). Συμφωνούμε, επίσης, μετά βεβαιότητας, ότι οι βασικές αρχές που περιέγραψα πιο πάνω, είναι βασικές για ένα δημοκρατικό κράτος σαν το Περού. Έτσι, θα μας ενδιέφερε πολύ να μάθουμε για οποιονδήποτε άλλο τρόπο διασφάλισης αυτών των αρχών, εκτός της χρήσης Ελεύθερου Λογισμικού, πάντοτε υπό τους όρους που περιγράφονται στο σχέδιο νόμου.

Όσο για τις παρατηρήσεις που κάνατε, θα προβούμε, τώρα, στη λεπτομερή ανάλυσή τους:

Κατ' αρχάς, σημειώνετε ότι: 1. "Η πρότασή σας επιβάλλει η κάθε Δημόσια υπηρεσία να χρησιμοποιεί κατ' αποκλειστικότητα Ελεύθερο Λογισμικό, γνωστό επίσης και σαν *Λογισμικό Ανοικτού Κώδικα*.

Αυτό είναι κάτι που παραβιάζει τις αρχές της ισότητας ενώπιον του Νόμου, της μη διάκρισης, της ελεύθερης ιδιωτικής πρωτοβουλίας, και τις ελευθερίες της βιομηχανίας και της σύναψης συμβάσεων (*industry and contracting*), οι οποίες προστατεύονται από το Σύνταγμα. "

Αυτή η αντίληψη είναι εσφαλμένη. Το νομοσχέδιο δεν επηρεάζει με κανένα τρόπο τα δικαιώματα που αναφέρετε. Περιορίζεται απολύτως στον καθορισμό των όρων για την χρήση του λογισμικού εκ μέρους κρατικών οργανισμών, χωρίς καθόλου να αναμειγνύεται στις συναλλαγές του

ιδιωτικού τομέως. Σύμφωνα με μία γενικώς αποδεκτή αρχή, το Κράτος δεν απολαμβάνει το ευρύ φάσμα ελευθεριών κατά τη σύναψη συμβάσεων, όπως εκείνη που έχει ο ιδιωτικός τομέας, καθώς οι δράσεις του περιορίζονται, ακριβώς λόγω των απαιτήσεων για διαφάνεια στις δημόσιες πράξεις και, κατ' αυτή την έννοια, η διαφύλαξη του γενικότερου κοινού συμφέροντος θα πρέπει να υπερισχύσει, όταν πρόκειται να νομοθετήσουμε για τα θέματα αυτά.

Το νομοσχέδιο προστατεύει την ισότητα ενώπιον του νόμου, καθώς δεν αποκλείεται κανένα φυσικό ή νομικό πρόσωπο από το δικαίωμα να υποβάλλει προσφορά για τα αγαθά αυτά προς το Κράτος, πάντοτε υπό τους όρους που ορίζει το νομοσχέδιο και χωρίς επιπλέον περιορισμούς πέραν εκείνων που ορίζει ο Νόμος του κράτους περί Συμβάσεων και Προμηθειών (TUO by Supreme Decree No. 012-2001-PCM).

Το νομοσχέδιο δεν εισάγει καμία διάκριση οποιουδήποτε είδους, αφού ορίζει μόνο το *πως* τα αγαθά αυτά πρέπει να παρέχονται (πράγμα που εμπίπτει στις εξουσίες του Κράτους) και όχι το *ποιος* πρέπει να τα παρέχει (πράγμα που θα αποτελούσε πραγματικά μία διάκριση, αν επιβάλλονταν περιορισμοί με βάση την εθνική προέλευση, την φυλή, τη θρησκεία, την ιδεολογία, την σεξουαλική προτίμηση, κλπ). Αντιθέτως, το νομοσχέδιο είναι σαφώς κατά των διακρίσεων. Αυτό συμβαίνει διότι με τον καθορισμό, χωρίς περιθώρια αμφιβολιών, των όρων για την παροχή λογισμικού, προλαμβάνει τις κρατικές υπηρεσίες από το να χρησιμοποιούν λογισμικό με μία άδεια που ενσωματώνει μεροληπτικούς όρους.

Θα πρέπει να κατέστη προφανές από τις προηγούμενες δύο παραγράφους ότι το νομοσχέδιο δεν βλάπτει τις ελεύθερες ιδιωτικές επιχειρήσεις, αφού αυτές οι τελευταίες μπορούν να επιλέγουν σε κάθε περίπτωση υπό ποιες συνθήκες θα παράγουν λογισμικό. Μερικά από αυτά θα είναι αποδεκτά από το Κράτος, ενώ άλλα όχι, αφού αντιτίθενται στην εγγύηση των βασικών αρχών που εξετέθησαν παραπάνω. Αυτή η ελεύθερη πρωτοβουλία είναι ασφαλώς συμβατή με την ελευθερία της βιομηχανίας (*freedom of industry*) και την ελευθερία σύναψης συμβάσεων (έστω και στην περιορισμένη μορφή στην οποία το Κράτος μπορεί να ασκεί αυτή την τελευταία). Οποιοσδήποτε ιδιώτης μπορεί, είτε να παράγει λογισμικό, υπό τους όρους που το Κράτος απαιτεί, είτε να απέχει από το να το πράξει. Κανένας δεν υποχρεώνεται να υιοθετήσει ένα μοντέλο παραγωγής, αλλά αν επιθυμούν να παρέχουν λογισμικό προς το Κράτος, τότε πρέπει να προσφέρουν τους μηχανισμούς που να εγγυώνται τις βασικές αρχές και που είναι εκείνες που περιγράφονται στο νομοσχέδιο.

Δίκην παραδείγματος: τίποτε μέσα στο κείμενο του νομοσχεδίου δεν θα μπορούσε να εμποδίσει την εταιρεία σας από το να προσφέρει σε Κρατικούς οργανισμούς μία σουίτα γραφείου, υπό τους όρους που ορίζονται στο νομοσχέδιο, και να καθορίσει την τιμή που κρίνετε ικανοποιητική. Εάν δεν το πράξετε, τότε δεν θα οφείλεται σε περιορισμούς που επιβάλλει ο νόμος, αλλά σε επιχειρηματικές αποφάσεις σχετικές με την μέθοδο προώθησης των προϊόντων σας, αποφάσεις στις οποίες το Κράτος δεν εμπλέκεται.

Επιπλέον, σημειώνετε ότι: 2. "Η πρότασή σας, επιβάλλοντας τη χρήση *Λογισμικού Ανοικτού Κώδικα*, ξεκινά μία μεταχείριση μη-ανταγωνιστική και με διακρίσεις, στη σύναψη συμβάσεων και προμηθειών από τους Δημόσιους οργανισμούς...".

Αυτή η δήλωση αποτελεί, απλώς, μία επανάληψη της προηγούμενης και, έτσι, η απάντηση μπορεί να εντοπισθεί πιο πάνω. Εν πάσει περιπτώσει, ας επικεντρωθούμε για μία στιγμή στο σχόλιό σας σχετικό με τις "μη-ανταγωνιστικές πρακτικές."

Ασφαλώς, στον καθορισμό οποιασδήποτε προμήθειας, ο αγοραστής καθορίζει τους όρους που σχετίζονται με την επιθυμητή χρήση του αγαθού ή της υπηρεσίας. Αυτό αποκλείει εξ' αρχής σε

ορισμένους κατασκευαστές να μπορούν να ανταγωνισθούν, αλλά δεν τους αποκλείει *εκ των προτέρων*, αλλά περισσότερο διότι βασίζεται σε μία σειρά αρχών που καθορίζονται από την αυτόνομη θέληση του αγοραστή και, έτσι, η διαδικασία λαμβάνει χώρα σύμφωνα με τον Νόμο. Στο νομοσχέδιο αυτό καθορίζεται ότι κανένας δεν αποκλείεται από τον ανταγωνισμό, στο βαθμό που εγγυάται την εκπλήρωση των βασικών αρχών.

Επιπλέον, το νομοσχέδιο *τονώνει* τον ανταγωνισμό, αφού τείνει να οδηγήσει προς μία προσφορά λογισμικού με καλύτερους όρους χρηστικότητας και στη βελτίωση των υπάρχουσών εργασιών, σε ένα περιβάλλον συνεχούς βελτίωσης.

Από την άλλη πλευρά, το κεντρικό θέμα του ανταγωνισμού είναι η ευκαιρία προσφοράς καλύτερων επιλογών στον καταναλωτή. Τώρα, είναι αδύνατον να αγνοήσουμε το γεγονός ότι το marketing δεν παίζει ένα ουδέτερο ρόλο, όταν το προϊόν προσφέρεται στην αγορά (αφού αν δεχθούμε το αντίθετο θα μας οδηγούσε να υποθέσουμε ότι τα έξοδα που κάνει η επιχείρηση για το marketing στερούνται οποιουδήποτε νοήματος), και, άρα, μία σημαντική δαπάνη σε αυτό το θέμα θα μπορούσε να επηρεάσει τις αποφάσεις του αγοραστή. Αυτή η επιρροή του marketing, μειώνεται, εν πολλοίς, από το Νόμο που υποστηρίζουμε, αφού η επιλογή, εντός του προτεινόμενου πλαισίου, βασίζεται στα *τεχνικά πλεονεκτήματα* του προϊόντος και όχι στην προσπάθεια που επενδύθηκε για την προώθησή του από τον παραγωγό. Υπό αυτή την έννοια, αυξάνεται ο ανταγωνισμός, αφού ακόμη και ο μικρότερος κατασκευαστής λογισμικού μπορεί να συναγωνισθεί επ' ίσοις όροις με τις πλέον ισχυρές πολυεθνικές.

Είναι απαραίτητο να τονίσουμε ότι δεν υπάρχει άποψη περισσότερο εναντίον του ανταγωνισμού από εκείνη των μεγάλων παραγωγών λογισμικού, που καταχρώνται συχνά την κυρίαρχη θέση τους, αφού, σε πολυάριθμες περιπτώσεις, προτείνουν ως λύση στα διάφορα προβλήματα που εντόπισαν οι χρήστες, τη συμβουλή "ενημερώστε το λογισμικό σας στη νέα έκδοση" (με έξοδα του χρήστη, φυσικά). Περαιτέρω, είναι συχνό να παρατηρείται μία αυθαίρετη διακοπή της τεχνικής βοήθειας για προϊόντα που, κατά την κρίση του κατασκευαστή και μόνο, είναι *πεπαλαιωμένα*. Κατ' αυτό τον τρόπο, για να πάρει οποιοδήποτε είδος τεχνικής βοήθειας, ο χρήστης βρίσκεται υποχρεωμένος να μεταβεί σε νέες εκδόσεις (με καθόλου ευκαταφρόνητα έξοδα, ειδικά καθώς εμπλέκονται, συχνά, και αλλαγές πλατφόρμας hardware). Καθώς, μάλιστα, όλη η υποδομή βασίζεται σε ιδιοταγείς μορφές αρχείου, ο χρήστης παραμένει *εγκλωβισμένος* στην ανάγκη για συνεχή χρήση προϊόντων από τον ίδιο προμηθευτή, ή θα πρέπει να καταβάλλει μία τεράστια προσπάθεια να αλλάξει σε ένα άλλο περιβάλλον (πιθανώς, εξ' ίσου ιδιοταγές).

Προσθέτετε: 3. "Αναγκάζοντας το Κράτος να προτιμήσει ένα επιχειρηματικό μοντέλο με αποκλειστική υποστήριξη του *Λογισμικού Ανοικτού Κώδικα*, η πρότασή σας θα αποθαρρύνει τους εντόπιους αλλά και τους διεθνείς κατασκευαστές λογισμικού, οι οποίοι προωθούν πραγματικές και σημαντικές επενδύσεις στη χώρα, δημιουργούν ένα σημαντικό αριθμό άμεσων και έμμεσων θέσεων εργασίας, και συνεπώς συνεισφέρουν στα εθνικά έσοδα. *Αντιθέτως*, οι τρόποι ανάπτυξης του *Λογισμικού Ανοικτού Κώδικα* αποφέρουν μικρότερα οφέλη για την οικονομία, αφού δημιουργούν κυρίως θέσεις στον τομέα των υπηρεσιών. "

Δε συμφωνώ με τον ισχυρισμό σας. Εν μέρει, χάρη σε εκείνο που εσείς ο ίδιος τονίζετε στην παράγραφο 6 της επιστολής σας, αναφορικά με το σχετικό βάρος των υπηρεσιών στα πλαίσια της χρήσης λογισμικού. Αυτή η αντίφαση, από μόνη της, αρκεί για να ακυρώσει το επιχειρήμά σας. Το μοντέλο υπηρεσιών που υιοθετήθηκε από πολυάριθμες εταιρείες στη βιομηχανία λογισμικού, είναι κατά πολύ μεγαλύτερο, από οικονομικής απόψεως, και με τάση για περαιτέρω αύξηση, από ότι το μοντέλο με αδειοδότηση των προγραμμάτων.

Από την άλλη πλευρά, ο ιδιωτικός τομέας της οικονομίας έχει τη μεγαλύτερη δυνατή ελευθερία επιλογής του οικονομικού μοντέλου που ταιριάζει καλύτερα στα συμφέροντά του, έστω και αν αυτή η ελευθερία επιλογής, συχνά, επισκιάζεται υποσυνείδητα από τη δυσανάλογη δαπάνη για το marketing εκ μέρους των παραγωγών ιδιοταγούς λογισμικού.

Επιπρόσθετα, μία ανάγνωση των απόψεών σας θα οδηγούσε στο συμπέρασμα ότι η Κρατική αγορά είναι ουσιώδους και κρίσιμης σημασίας για τη βιομηχανία ιδιοταγούς λογισμικού, σε τέτοιο σημείο ώστε η επιλογή που έκανε το Κράτος σε αυτό το σχέδιο νόμου θα μπορούσε να εξαφανίσει εντελώς την αγορά για τις βιομηχανίες αυτές. Αν αυτό αληθεύει, μπορούμε να συμπεράνουμε ότι το Κράτος θα πρέπει να επιδοτεί τη βιομηχανία ιδιοταγούς λογισμικού. Στην απίθανη περίπτωση που κάτι τέτοιο ήταν αληθές, το Κράτος θα είχε το δικαίωμα να εφαρμόσει τις επιδοτήσεις στον τομέα που θεωρεί ότι έχει τη μεγαλύτερη κοινωνική αξία. Είναι αναντίρρητο ότι, σε αυτή την απίθανη περίπτωση, αν το Κράτος αποφάσιζε να επιδοτήσει το λογισμικό, θα έπρεπε να επιλέξει το Ελεύθερο Λογισμικό, αντί για το ιδιοταγές, λαμβάνοντας υπ' όψη τον κοινωνικό του αντίκτυπο και την ορθολογική χρήση των χρημάτων του φορολογουμένου.

Αναφορικά με τις θέσεις εργασίας που δημιουργήθηκαν από το ιδιοταγές λογισμικό, σε χώρες σαν τη δική μας, αυτές αφορούν κυρίως τεχνικές ικανότητες, με μικρή προστιθέμενη αξία. Σε τοπικό επίπεδο, οι τεχνικοί που παρέχουν υποστήριξη για ιδιοταγές λογισμικό, που παράγεται από πολυεθνικές εταιρείες, δεν έχουν την δυνατότητα να διορθώνουν *bugs*, όχι απαραίτητα λόγω έλλειψης τεχνικής ικανότητας ή ταλέντου, αλλά διότι δεν έχουν πρόσβαση στον πηγαίο κώδικα για να τα διορθώσουν. Με το Ελεύθερο Λογισμικό μπορεί κάποιος να δημιουργήσει ένα τεχνικώς περισσότερο καταρτισμένο δυναμικό προς απασχόληση, καθώς και ένα πλαίσιο για ελεύθερο ανταγωνισμό, όπου η επιτυχία συνδέεται μόνο με την ικανότητα προσφοράς καλής τεχνικής υποστήριξης και ποιοτικής παροχής υπηρεσιών. Μπορεί, επίσης, να τονώσει την αγορά, και μπορεί να αυξήσει το κοινό αποθετήριο γνώσης, ανοίγοντας εναλλακτικές για μία προσφορά υπηρεσιών με μεγαλύτερη συνολική προστιθέμενη αξία και υψηλότερου ποιοτικού επιπέδου, προς όφελος όλων των εμπλεκόμενων: των παραγωγών, των οργανώσεων παροχής υπηρεσιών και των καταναλωτών.

Στις αναπτυσσόμενες χώρες, αποτελεί πολύ συχνό φαινόμενο για τις εντόπιες βιομηχανίες λογισμικού να κερδίζουν τις περισσότερες αναθέσεις συμβάσεων στον τομέα των υπηρεσιών, είτε στη δημιουργία λογισμικού *ad hoc* (επί παραγγελία). Συνεπώς, οποιαδήποτε αρνητική επίπτωση και αν έχει η εφαρμογή του νομοσχεδίου σε αυτό τον τομέα, θα αντισταθμίζεται παραπάνω από ικανοποιητικά από την αύξηση της ζήτησης σε υπηρεσίες (εφ' όσον αυτές προσφέρονται σε υψηλά ποιοτικά επίπεδα). Αν οι πολυεθνικές εταιρείες λογισμικού αποφασίσουν να μην ανταγωνισθούν κάτω από αυτούς τους νέους κανόνες του παιγνιδιού, τότε είναι πιθανόν να υποστούν κάποια μείωση των κερδών τους, από την άποψη των εισπράξεων από τις άδειες. Εν πάσει περιπτώσει, λαμβάνοντας υπ' όψη ότι αυτές οι επιχειρήσεις συνεχίζουν να υποστηρίζουν ότι μεγάλο μέρος του λογισμικού που χρησιμοποιεί το Κράτος έχει αντιγραφεί παρανόμως, τότε θα μπορούσε κάποιος να προβλέψει ότι η επίπτωση δεν θα είναι εν τέλει και πολύ σοβαρή. Σίγουρα, σε οποιαδήποτε περίπτωση, η τύχη τους θα καθορισθεί από τους νόμους της αγοράς, οι αλλαγές των οποίων δεν μπορούν να αποφευχθούν. Πολλές επιχειρήσεις που συνδέονται παραδοσιακά με ιδιοταγές λογισμικό, έχουν ήδη δρομολογήσει (στηριζόμενες και σε τεράστια έξοδα) την παροχή υπηρεσιών που συνδέονται με Ελεύθερο Λογισμικό, πράγμα που δείχνει ότι τα δύο μοντέλα δεν αλληλο-αποκλείονται αμοιβαία.

Με αυτό το Νομοσχέδιο το Κράτος αποφασίζει ότι πρέπει να διαφυλάξει ορισμένες βασικές αξίες. Αποφασίζει, λοιπόν, κάτι τέτοιο βασιζόμενο στις κυρίαρχες εξουσίες του, χωρίς να θίγει κατά κανένα τρόπο τις συνταγματικές εγγυήσεις. Αν αυτές οι αξίες μπορούσαν να διασφαλισθούν, χωρίς να έπρεπε να επιλέξουμε ένα ιδιαίτερο οικονομικό μοντέλο, τα αποτελέσματα του νόμου θα

απέβαιναν ακόμη πιο επωφελή. Σε κάθε περίπτωση, θα έπρεπε να είναι σαφές ότι το Κράτος δεν επιλέγει ένα οικονομικό μοντέλο. Αν συμβαίνει να υπάρχει μόνον ένα οικονομικό μοντέλο, ικανό να προσφέρει λογισμικό που να παρέχει τη βασική εγγύηση για αυτές τις αξίες, αυτό είναι χάρη σε κάποιες ιστορικές περιστάσεις και όχι λόγω μίας αυθαίρετης επιλογής ενός δεδομένου μοντέλου.

Η επιστολή σας συνεχίζει: "4. "Η πρότασή σας επιβάλλει τη χρήση Ανοικτού Λογισμικού χωρίς να λαμβάνει υπ' όψη τους κινδύνους που αυτό συνεπάγεται για την ασφάλεια, την εξασφάλιση, ακόμη και την πιθανή παραβίαση των δικαιωμάτων πνευματικής ιδιοκτησίας τρίτων."

Υπαινισσόμενος κατά έναν αφηρημένο τρόπο "τους κινδύνους που αυτό συνεπάγεται", χωρίς να αναφέρεσθε εξειδικευμένα ούτε σε έναν από αυτούς τους υποτιθέμενους κινδύνους, δείχνει τουλάχιστον μία κάποια έλλειψη γνώσης του θέματος. Γι' αυτό, επιτρέψτε μου να σας διαφωτίσω σε αυτά τα σημεία.

Σχετικά με την ασφάλεια:

Στο αρχικό κομμάτι της συζήτησης περί των βασικών αρχών του νομοσχεδίου, αναφέρθηκαν ήδη, σε γενικές γραμμές, τα θέματα εθνικής ασφάλειας. Πιο συγκεκριμένα, σχετικά με την ασφάλεια του ίδιου του λογισμικού, είναι πασίγνωστο ότι όλα τα λογισμικά (είτε ιδιοταγή, είτε ελεύθερα) περιέχουν σφάλματα ή bugs (στην κοινή ιδιόλεκτο των προγραμματιστών). Είναι, όμως, επίσης γνωστό ότι τα bugs στο ελεύθερο λογισμικό είναι λιγότερα και διορθώνονται πολύ γρηγορότερα από ότι στο ιδιοταγές λογισμικό. Δεν είναι τυχαίο ότι πολυάριθμοι δημόσιοι οργανισμοί, υπεύθυνοι για την ασφάλεια των κρατικών συστημάτων IT σε ανεπτυγμένες χώρες, απαιτούν την χρήση ελεύθερου λογισμικού όταν δέχονται τέτοιες απαιτήσεις για ασφάλεια και αποτελεσματικότητα.

Εκείνο που είναι αδύνατον να αποδειχθεί είναι ότι το ιδιοταγές λογισμικό είναι πιο ασφαλές από το ελεύθερο, χωρίς τον δημόσιο και ανοιχτό έλεγχο εκ μέρους της επιστημονικής κοινότητας και των ίδιων των χρηστών. Αυτού του είδους η απόδειξη είναι αδύνατη, διότι το ίδιο το μοντέλο του ιδιοταγούς λογισμικού εμποδίζει μία τέτοια ανάλυση, οπότε, οποιαδήποτε εγγύηση περί ασφάλειας επαφίεται μόνο στις υποσχέσεις περί καλών προθέσεων (μεροληπτικές, με ότι κριτήρια και αν τις δούμε) που παρέχει ο ίδιος ο κατασκευαστής, ή οι συνεργάτες του.

Θα πρέπει να υπενθυμίζεται ότι, σε πολλές περιπτώσεις, οι όροι αδειοδότησης περιλαμβάνουν ρήτρες περί μη αποκάλυψης, που εμποδίζουν τον χρήστη να προβεί σε δημόσια αποκάλυψη των σφαλμάτων ασφαλείας που εντοπίζει μέσα στο ιδιοταγές αδειοδοτημένο προϊόν.

Αναφορικά με την εγγύηση:

Όπως γνωρίζετε πολύ καλά, ή, εν πάσει περιπτώσει, όπως μπορείτε να βρείτε με μία απλή ανάγνωση της EULA (*End User License Agreement*) των προϊόντων που αδειοδοτείτε, στη μεγάλη πλειοψηφία των περιπτώσεων, οι εγγυήσεις περιορίζονται στην αντικατάσταση του μέσου αποθήκευσης σε περίπτωση που αυτό παρουσιάσει μειονεκτήματα, αλλά σε καμία περίπτωση δεν προβλέπουν αποζημίωση έναντι άμεσων ή έμμεσων βλαβών, απωλεσθέντων κερδών, κλπ...

Αν, ως αποτέλεσμα ενός σφάλματος ασφαλείας (*security bug*), σε ένα από τα δικά σας προϊόντα, που δεν επισκευάσθηκε εν καιρώ από εσάς τους ίδιους, ένας εισβολέας καταφέρει να διεισδύσει σε κρίσιμα Κρατικά συστήματα, σε τι είδους εγγυήσεις, αποκαταστάσεις και αποζημιώσεις θα προέβαινε η εταιρεία σας σύμφωνα με τους όρους της αδείας σας; Οι εγγυήσεις του ιδιοταγούς λογισμικού, ως προς το ότι τα προγράμματα διανέμονται "ως έχουν", δηλαδή, στην ακριβή κατάσταση στην οποία βρίσκονται, χωρίς ανάληψη πρόσθετης ευθύνης εκ μέρους του προμηθευτή σχετικά με την

λειτουργία, δεν διαφέρουν σε τίποτε από τις εγγυήσεις που συνήθως δίδονται με το Ελεύθερο Λογισμικό.

Σχετικά με την Πνευματική Ιδιοκτησία:

Τα θέματα που άπτονται της πνευματικής ιδιοκτησίας, βρίσκονται εκτός του πεδίου του παρόντος νομοσχεδίου, από τη στιγμή που καλύπτονται από άλλους ειδικούς νόμους. Το μοντέλο του Ελεύθερου Λογισμικού δεν υπονοεί σε καμία περίπτωση άγνοια αυτών των νόμων και, πραγματικά, η μεγάλη πλειοψηφία του Ελεύθερου Λογισμικού καλύπτεται από τα δικαιώματα πνευματικής ιδιοκτησίας (*copyright*). Στην πραγματικότητα, η ενσωμάτωση αυτού του θέματος στις παρατηρήσεις σας δείχνει τη σύγχυσή σας γύρω από το νομικό πλαίσιο, εντός του οποίου αναπτύχθηκε το Ελεύθερο Λογισμικό. Η συμπερίληψη της πνευματικής ιδιοκτησίας άλλων, μέσα σε έργα που ισχυριζόμαστε ότι είναι δικά μας, δεν αποτελεί πρακτική που συνηθίζεται στην κοινότητα του Ελεύθερου Λογισμικού. Ενώ, δυστυχώς, χαρακτήρισε τον τομέα του ιδιοταγούς λογισμικού. Για παράδειγμα, η καταδίκη της Microsoft Corp., στις 27 Σεπτεμβρίου του 2001, από το Εμπορικό Δικαστήριο (Commercial Court) της πόλης της Nanterre, Γαλλία, με ένα πρόστιμο 3 εκατομ. Φράγκων για ζημίες και τόκους, λόγω παραβίασης της πνευματικής ιδιοκτησίας (“πειρατεία”, για να χρησιμοποιήσουμε τον ατυχή όρο που η εταιρεία σας συχνά χρησιμοποιεί στις διαφημίσεις της).

Συνεχίζετε λέγοντας ότι: 5. "Προϋποθέτει, εσφαλμένα, ότι το *Λογισμικό Ανοικτού Κώδικα* είναι Δωρεάν Λογισμικό, δηλαδή χωρίς κόστος, φθάνοντας έτσι σε λάθος συμπεράσματα για το πόσα χρήματα θα εξοικονομήσει το Κράτος. Δεν προσφέρει μία ανάλυση κόστους/ οφέλους για να υποστηρίξει αυτή την υπόθεση."

Αυτή η παρατήρηση είναι εσφαλμένη. Επί της αρχής, η ελευθερία και η έλλειψη κόστους είναι ορθογώνιες έννοιες: υπάρχει λογισμικό που είναι ιδιοταγές και για το οποίο πληρώνει κανείς (για παράδειγμα, το MS Office), λογισμικό που είναι ιδιοταγές αλλά χωρίς τίμημα (το MS Internet Explorer), λογισμικό που είναι Ελεύθερο αλλά για το οποίο πληρώνει κανείς (Red Hat, SuSE κλπ., διανομές Gnu/Linux), λογισμικό που είναι Ελεύθερο και δωρεάν (πχ. το Apache, το OpenOffice, ο Mozilla), αλλά ακόμη και λογισμικό που μπορεί να κυκλοφορεί με διάφορους συνδυασμούς αδειών (MySQL).

Σίγουρα, “Ελεύθερο Λογισμικό” δεν σημαίνει απαραίτητως και “δωρεάν”. Ούτε το κείμενο του νομοσχεδίου δηλώνει ότι πρέπει να είναι έτσι τα πράγματα, όπως θα προσέξατε και εσείς, μετά την ανάγνωση του. Οι ορισμοί που περιλαμβάνονται στο νομοσχέδιο αναφέρουν σαφώς τι θα πρέπει να θεωρείται Ελεύθερο Λογισμικό, χωρίς να γίνεται αναφορά σε κανένα σημείο στην έννοια του δωρεάν. Αν και αναφέρεται η δυνατότητα επίτευξης εξοικονόμησης σε πληρωμές για άδειες ιδιοταγών λογισμικών, τα θεμέλια του νομοσχεδίου στηρίζονται σαφώς στις ουσιώδεις εγγυήσεις που πρέπει να διαφυλάσσονται και στα ερεθίσματα για την τοπική τεχνολογική ανάπτυξη. Με δεδομένο ότι ένα δημοκρατικό κράτος πρέπει να στηρίζει αυτές τις αρχές, δεν έχει άλλη επιλογή από τη χρήση λογισμικού που να διαθέτει δημοσίως ελεύθερο τον πηγαίο του κώδικα και να μπορεί να ανταλλάσσει πληροφορίες μόνο με τις προτυποποιημένες μορφές (*standard formats*).

Αν το Κράτος δεν χρησιμοποιεί λογισμικό με αυτά τα χαρακτηριστικά, θα αποδυναμώνει βασικές δημοκρατικές αρχές. Ευτυχώς, η χρήση του Ελεύθερου Λογισμικού συνεπάγεται και χαμηλότερα συνολικά κόστη. Ακόμη, όμως, και αν υποθέταμε ότι ήταν περισσότερο δαπανηρό σε σχέση με το ιδιοταγές λογισμικό (πράγμα που εύκολα μπορεί να καταρριφθεί), η απλή ύπαρξη και μόνον ενός αποτελεσματικού εργαλείου Ελεύθερου Λογισμικού, για μία οποιαδήποτε λειτουργία IT, θα υποχρέωνε το Κράτος να το χρησιμοποιήσει. Όχι, λοιπόν, με εντολή αυτού Νομοσχεδίου, αλλά με

βάση τις ουσιώδεις αρχές που εκθέσαμε στην αρχή και οι οποίες προκύπτουν από αυτή καθ' εαυτή τη φύση ενός έννομου δημοκρατικού Κράτους.

Συνεχίζετε περαιτέρω: 6. "Είναι εσφαλμένη η σκέψη ότι το *Λογισμικό Ανοικτού Κώδικα* είναι δωρεάν. Μία έρευνα που έκαναν οι Gartner Group (σημαντικοί και διεθνώς γνωστοί ερευνητές της αγοράς σε θέματα του κόσμου της τεχνολογίας), έδειξε ότι το κόστος κτήσης του λογισμικού (λειτουργικό σύστημα και εφαρμογές) είναι μόνο το 8% του συνολικού κόστους ιδιοκτησίας που οι επιχειρήσεις και οι οργανισμοί πρέπει να αντιμετωπίσουν, ως συνέπεια της ορθολογικής και παραγωγικής χρήσης της τεχνολογίας." Το υπόλοιπο 92% αντιπροσωπεύει κόστη εγκατάστασης, εκπαίδευσης, υποστήριξης, συντήρησης, διαχείρισης και downtime."

Αυτό το επιχείρημα επαναλαμβάνει εκείνο που παρουσιάστηκε, ήδη, στην παράγραφο 5 και, εν μέρει, αντιφάσκει με την παράγραφο 3. Προς χάριν συντομίας, παραπέμπουμε στα σχόλια που υπάρχουν σε εκείνες τις παραγράφους. Εν πάσει περιπτώσει, επιτρέψτε μου να υποδείξω ότι το συμπέρασμά σας, από την άποψη της λογικής, είναι εσφαλμένο: ακόμη και αν, σύμφωνα με την ομάδα Gartner Group, το κόστος του λογισμικού είναι, κατά μέσον όρο, μόνο το 8% του συνολικού κόστους χρήσης, αυτό δεν αποκλείει, κατά κανένα τρόπο, την ύπαρξη λογισμικού χωρίς κόστος, δηλαδή, με μηδενικό κόστος αδειοδότησης.

Επιπροσθέτως, σε αυτή την παράγραφο, επισημαίνετε, ορθώς, ότι τα στοιχεία που σχετίζονται με τις υπηρεσίες και τις απώλειες που οφείλονται σε down time, συνιστούν το μεγαλύτερο μέρος του συνολικού κόστους χρήσης του λογισμικού, πράγμα που, όπως θα παρατηρήσατε, ακυρώνει την θέση σας αναφορικά με τη μικρή αξία των υπηρεσιών, που αναφέρεται στην παράγραφο 3. Τώρα, η χρήση Ελεύθερου Λογισμικού, συνεισφέρει σημαντικά στη μείωση κόστους του υπόλοιπου κύκλου ζωής. Αυτή η μείωση του κόστους εγκατάστασης, της υποστήριξης κλπ., μπορεί να γίνει αισθητή σε πολλούς τομείς:

κατ' αρχάς, στο ανταγωνιστικό μοντέλο παροχής υπηρεσιών του Ελεύθερου Λογισμικού, η υποστήριξη και η συντήρηση οι οποίες μπορούν ελεύθερα να δοθούν σε μία σειρά προμηθευτών που θα ανταγωνίζονται με βάση την ποιότητα και το χαμηλό κόστος (αυτό αληθεύει για την εγκατάσταση, την ενεργοποίηση και την υποστήριξη, καθώς και μεγάλο μέρος της συντήρησης). Κατά δεύτερο λόγο, χάρη στα χαρακτηριστικά αναπαραγωγής του μοντέλου, η συντήρηση που εκτελείται για μία εφαρμογή, είναι εύκολα αντιγράψιμη, χωρίς να εξαναγκάζεται κανείς σε μεγάλα έξοδα (δηλαδή, χωρίς να χρειάζεται να πληρώσει περισσότερο από μία φορά, για το ίδιο πράγμα) αφού οι τροποποιήσεις, εφ' όσον κάποιος το επιθυμεί, μπορούν να ενσωματωθούν στον κοινό κορμό της γνώσης. Τρίτο, τα τεράστια κόστη που προκαλούνται από το λογισμικό που δεν λειτουργεί (οι λεγόμενες *blue screens of death*, κακόβουλος κώδικας όπως ιοί, σκουλήκια/ *worms*, και δούρειοι ίπποι/ *trojans*, εξαιρέσεις, γενικά σφάλματα προστασίας και άλλα, πολύ γνωστά προβλήματα) μειώνονται σημαντικά με τη χρήση πιο σταθερού λογισμικού. Είναι, εξ' άλλου, πασίγνωστο ότι μία από τις πιο αξιοσημείωτες αρετές του Ελεύθερου Λογισμικού είναι η σταθερότητα.

Αναφέρετε περαιτέρω ότι: 7. "Ένα από τα επιχειρήματα που στηρίζουν την πρότασή σας είναι η υποτιθέμενη φθηνότητα του λογισμικού ανοικτού κώδικα σε σύγκριση με το εμπορικό λογισμικό, χωρίς να λαμβάνεται υπ' όψη η δυνατότητα των μοντέλων μαζικής αδειοδότησης (volume licensing). Το Κράτος μπορεί πραγματικά να επωφεληθεί από αυτά, όπως και σε άλλες χώρες."

Όπως έχω ήδη τονίσει, εκείνο που είναι σημαντικό δεν είναι το κόστος του λογισμικού, αλλά οι αρχές της ελευθερίας των πληροφοριών, της προσβασιμότητας και της ασφάλειας. Αυτά τα επιχειρήματα έχουν καλυφθεί εκτεταμένα στις προηγούμενες παραγράφους, προς τις οποίες θα σας παραπέμψω.

Από τη άλλη μεριά, ασφαλώς και υπάρχουν διάφοροι τύποι μαζικής αδειοδότησης (αν και, δυστυχώς, το ιδιοταγές λογισμικό δεν ικανοποιεί τις βασικές αρχές). Όπως, όμως, τονίσατε και εσείς, πολύ ορθώς, στην αμέσως προηγούμενη παράγραφο της επιστολής σας, καταφέρνουν μόνον να μειώσουν την επίπτωση μίας παραμέτρου που δεν αντιπροσωπεύει παραπάνω από το 8% του συνόλου.

Συνεχίζετε: 8. "Επιπροσθέτως, η προσέγγιση που επιλέχθηκε από το δικό σας project (i) είναι σαφώς δαπανηρότερη, λόγω του υψηλού κόστους της μετάβασης, (ii) διακινδυνεύει την απώλεια διαλειτουργικότητας μεταξύ πληροφοριακών συστημάτων, τόσο εντός της Κρατικής μηχανής, όσο και ανάμεσα στο Κράτος και τον ιδιωτικό τομέα, λόγω των πολλών διαφορετικών διανομών του λογισμικού ανοικτού κώδικα που κυκλοφορούν στην αγορά."

Ας αναλύσουμε τις τοποθετήσεις σας σε δύο μέρη. Το πρώτο σας επιχείρημα, ότι η διαδικασία μετάβασης επιφέρει μεγάλα κόστη, αποτελεί, στην πραγματικότητα, ένα επιχείρημα υπέρ του νομοσχεδίου. Διότι, όσο περισσότερο περνά ο καιρός, τόσο πιο δύσκολη θα γίνεται και η μετάβαση προς μία άλλη τεχνολογία. Αλλά, συγχρόνως, οι κίνδυνοι που συνδέονται με το ιδιοταγές λογισμικό, θα συνεχίζουν να αυξάνονται. Κατ' αυτό τον τρόπο, η χρήση ιδιοταγών συστημάτων και προτύπων, θα κάνει το Κράτος ακόμη πιο εξαρτημένο από συγκεκριμένους προμηθευτές.

Αντιθέτως, άπαξ και υιοθετηθεί μία πολιτική χρήσης Ελεύθερου Λογισμικού (πράγμα που, σίγουρα, συνεπάγεται κάποιο κόστος), τότε, αντιθέτως, η μετάβαση από το ένα σύστημα στο άλλο, γίνεται πολύ απλή, αφού όλα τα δεδομένα είναι αποθηκευμένα σε ανοικτά πρότυπα. Από την άλλη μεριά, η μετάβαση προς ένα περιβάλλον ανοικτού λογισμικού δεν επιφέρει παραπάνω κόστη, από ότι η μετάβαση μεταξύ δύο διαφορετικών ιδιοταγών συστημάτων, πράγμα που ακυρώνει τελείως το επιχείρημά σας.

Το δεύτερο επιχείρημα αναφέρεται στην "απώλεια διαλειτουργικότητας μεταξύ πληροφοριακών συστημάτων, τόσο εντός της Κρατικής μηχανής, όσο και ανάμεσα στο Κράτος και τον ιδιωτικό τομέα".

Αυτή η τοποθέτηση υποδεικνύει μία κάποια άγνοια του τρόπου με τον οποίο κατασκευάζεται το Ελεύθερο Λογισμικό, το οποίο δε μεγιστοποιεί την εξάρτηση του χρήστη από καμία συγκεκριμένη πλατφόρμα, όπως, συνήθως, συμβαίνει στο βασίλειο του ιδιοταγούς λογισμικού.

Ακόμη και όταν υπάρχουν πολλές διανομές Ελεύθερου Λογισμικού, ή και πολλά άλλα προγράμματα που μπορούν να χρησιμοποιηθούν για την ίδια λειτουργία, η διαλειτουργικότητα είναι εγγυημένη, τόσο από την ίδια τη χρήση προτυποποιημένων μορφών (*standard formats*), όπως ακριβώς απαιτεί και το νομοσχέδιο, όσο και από τη δυνατότητα δημιουργίας διαλειτουργικού λογισμικού, εφόσον είναι διαθέσιμος ο πηγαίος κώδικας.

Κατόπιν, λέτε: 9. "Στις περισσότερες περιπτώσεις, το λογισμικό ανοικτού κώδικα δεν προσφέρει ικανοποιητικά επίπεδα υπηρεσιών για να μπορέσουν οι χρήστες του να πετύχουν μία αυξημένη παραγωγικότητα, ούτε προσφέρει εγγυήσεις από γνωστούς κατασκευαστές. Αυτά τα πράγματα οδήγησαν πολλούς δημόσιους οργανισμούς να ανακαλέσουν τις αποφάσεις τους περί χρήσης λογισμικού Ανοικτού Κώδικα. Τώρα, στη θέση του, χρησιμοποιούν εμπορικό λογισμικό".

Αυτή η παρατήρηση στερείται παντελώς βασιμότητας. Αναφορικά με τις εγγυήσεις, το επιχείρημά σας απερρίφθη με την απάντηση στην παράγραφο 4. Αναφορικά με τις υπηρεσίες στήριξης, είναι δυνατόν να χρησιμοποιεί κανείς Ελεύθερο Λογισμικό ακόμη και χωρίς αυτές (όπως, ακριβώς,

συμβαίνει και με το ιδιοταγές λογισμικό), αλλά οποιοσδήποτε τις χρειάζεται, μπορεί να αγοράσει την υποστήριξη ξεχωριστά, είτε από τοπικές επιχειρήσεις, είτε από πολυεθνικές εταιρείες, ξανά, με τρόπο ανάλογο με αυτόν που παρατηρείται και με το ιδιοταγές λογισμικό.

Από την άλλη πλευρά, θα βοηθούσε τα μέγιστα στην ανάλυσή μας, αν μπορούσατε να μας ενημερώσετε έστω και για ένα παράδειγμα έργου Ελεύθερου Λογισμικού, που να έχει υιοθετηθεί σε μία Δημόσια Διοίκηση και που να έχει, ήδη, εγκαταλειφθεί, υπέρ του ιδιοταγούς λογισμικού. Εμείς γνωρίζουμε πολλές περιπτώσεις, στις οποίες συνέβη ακριβώς το αντίθετο, αλλά δεν μας προκύπτει καμία περίπτωση στην οποία να έλαβε χώρα αυτό που εσείς περιγράφετε.

Συνεχίζετε με την παρατήρηση ότι: 10. "Αυτό το project αποθαρρύνει τη δημιουργικότητα στην Περουβιανή βιομηχανία λογισμικού, που πωλεί αγαθά αξίας 40 εκατομμυρίων δολαρίων ετησίως, εκ των οποίων 4 εκατομμύρια δολάρια αντιστοιχούν σε εξαγωγές (10^η θέση στη κατάταξη των μη-παραδοσιακών εξαγωγών του Περού, παραπάνω από τα χειροποίητα αγαθά) και αποτελεί μία πηγή για θέσεις εργασίας υψηλής εξειδίκευσης. Με έναν νόμο που ενθαρρύνει τη χρήση λογισμικού Ανοικτού Κώδικα, οι προγραμματιστές χάνουν τα παραδοσιακά τους δικαιώματα πνευματικής ιδιοκτησίας και την πλέον σημαντική τους πηγή κερδών."

Είναι αρκούντως σαφές ότι κανείς δεν υποχρεώνεται να κυκλοφορήσει τον κώδικά του στο εμπόριο ως Ελεύθερο Λογισμικό. Το μόνο πράγμα που θα πρέπει να ληφθεί υπ' όψη είναι ότι, εάν δεν είναι Ελεύθερο Λογισμικό, τότε δε θα μπορεί να πωλείται στο Δημόσιο Τομέα. Αυτή όμως δεν είναι, σε καμία περίπτωση, η κύρια αγορά, για την εθνική βιομηχανία λογισμικού. Καλύψαμε, ήδη, μερικές ερωτήσεις, που αναφέρονται στην επιρροή του νομοσχεδίου για τη δημιουργία απασχόλησης, με θέσεις εργασίας που ήταν, συγχρόνως, και με υψηλή τεχνική εξειδίκευση, αλλά και υπό καλύτερες συνθήκες για περαιτέρω ανταγωνισμό, οπότε δεν έχει αξία να επιμείνουμε σε αυτό το σημείο.

Αυτό που ακολουθεί στην τοποθέτησή σας, είναι λάθος. Από τη μία μεριά, κανένας δημιουργός Ελεύθερου Λογισμικού δεν χάνει τα δικαιώματά του στην πνευματική ιδιοκτησία, εκτός και αν εκφράσει ρητώς τη βούλησή του να θέσει τη δουλειά του στη διάθεση του κοινού. Το κίνημα του Ελεύθερου Λογισμικού επέδειξε πάντοτε πολύ μεγάλο σεβασμό προς την πνευματική ιδιοκτησία και οδήγησε στη δημόσια και ευρύτατη αναγνώριση των συγγραφέων. Ονόματα όπως εκείνα των Richard Stallman, Linus Torvalds, Guido van Rossum, Larry Wall, Miguel de Icaza, Andrew Tridgell, Theo de Raadt, Andrea Arcangeli, Bruce Perens, Darren Reed, Alan Cox, Eric Raymond, και πολλών άλλων, είναι παγκοσμίως αναγνωρίσιμα για τις συνεισφορές τους στην ανάπτυξη λογισμικού, που χρησιμοποιείται σήμερα από εκατομμύρια ανθρώπους, σε παγκόσμια κλίμακα, ενώ υπάρχουν πολλοί δημιουργοί εξαιρετικών ιδιοταγών λογισμικών, που παραμένουν ανώνυμοι. Από την άλλη, η άποψη ότι οι αμοιβές για τα δικαιώματα των συγγραφέων αποτελούν την κύρια πηγή εισοδήματος των Περουβιανών προγραμματιστών, συνιστά, σε κάθε περίπτωση, μία εικασία, ειδικά επειδή δεν υπάρχει απόδειξη προς αυτή την κατεύθυνση, ούτε μία ένδειξη για το πως η χρήση Ελεύθερου Λογισμικού εκ μέρους του Κράτους θα μπορούσε να επηρεάσει αυτά τα εισοδήματα.

Συνεχίζετε με την τοποθέτηση ότι: 11. "Από τη στιγμή που το λογισμικό ανοικτού κώδικα μπορεί να διανέμεται ελεύθερα, δεν μπορεί να οδηγήσει καθόλου στη δημιουργία κερδών για τους προγραμματιστές του μέσω των εξαγωγών. Με αυτό τον τρόπο, εξασθενεί το πολλαπλασιαστικό αποτέλεσμα των πωλήσεων λογισμικού προς άλλες χώρες και ανακόπτει την πορεία ανάπτυξης της τοπικής βιομηχανίας, την οποία το Κράτος θα έπρεπε να υποβοηθά."

Αυτή η τοποθέτηση δείχνει, για μία ακόμη φορά, πλήρη άγνοια των μηχανισμών κα της αγοράς του Ελεύθερου Λογισμικού. Προσπαθεί να ισχυρισθεί ότι η αγορά που προκύπτει από την πώληση των μη-αποκλειστικών δικαιωμάτων χρήσης (δηλ. από την πώληση των Αδειών), συνιστά την μόνη εφικτή για τη βιομηχανία λογισμικού, ενώ εσείς ο ίδιος τονίσατε σε διάφορες από τις προηγούμενες

παραγράφους, ότι δεν είναι ούτε καν η πιο σημαντική. Τα κίνητρα που παρέχει ο Νόμος για την ανάπτυξη μίας ροής καλύτερα εξειδικευμένων επαγγελματιών, μαζί με την αύξηση σε εμπειρία που θα προκύψει μέσω της δυνατότητας που θα έχουν οι Περουβιανοί τεχνικοί να εργασθούν με Ελεύθερο Λογισμικό σε ευρεία κλίμακα, μέσα στα πλαίσια του Κράτους, θα τους θέσουν σε μία σαφώς πλεονεκτική ανταγωνιστική θέση, ούτως ώστε να μπορούν να προσφέρουν τις υπηρεσίες τους και στο εξωτερικό.

Κατόπιν, λέτε πως: "12. Στο forum, συζητήθηκε, επίσης, η σημασία της χρήσης του λογισμικού ανοικτού κώδικα στην Εκπαίδευση, χωρίς να γίνεται κανένα σχόλιο για την πλήρη αποτυχία αυτής της πρωτοβουλία, σε χώρες σαν το Μεξικό. Εκεί, οι ίδιοι οι αξιωματούχοι του Κράτους που υποστήριζαν το project, τώρα λένε ότι το λογισμικό ανοικτού κώδικα δεν προσέφερε μία εκπαιδευτική εμπειρία προς τα παιδιά των Σχολείων, δεν υπήρχαν κατάλληλα επίπεδα εκπαίδευσης σε πανεθνικό επίπεδο, παρέχονταν ακατάλληλη υποστήριξη προς την πλατφόρμα και το λογισμικό δεν ήταν επαρκώς ενσωματωμένο με τα υπάρχοντα υπολογιστικά Σχολικά συστήματα."

Πραγματικά, το Μεξικό προχώρησε με την όπισθεν με το έργο Red Escolar (Σχολικό Δίκτυο). Αυτό οφείλεται στο γεγονός ότι οι κινητήριες δυνάμεις που βρισκόταν πίσω από το Μεξικανικό project, χρησιμοποίησαν τα κόστη των αδειών χρήσεως ως το βασικό τους επιχείρημα, αντί για τους άλλους λόγους, που αναπτύξαμε στο δικό μας project, και οι οποίοι είναι, μακράν, οι πιο ουσιαστικοί. Λόγω αυτού του εννοιολογικού σφάλματος και, ως αποτέλεσμα της ελλειπούς πρακτικής υποστήριξης από το Υπουργείο Παιδείας, οδήγησε στην παραδοχή ότι, για να πλασάρει κανείς Ελεύθερο Λογισμικό στα Σχολεία, θα ήταν αρκετό να καταργηθεί ο προϋπολογισμός τους για λογισμικό και, απλώς, να τους αποσταλεί ένα CD ROM με το λειτουργικό σύστημα GNU/Linux. Προφανώς, αυτό απέτυχε και δεν θα μπορούσε να συμβεί διαφορετικά, ακριβώς με τον ίδιο τρόπο που απέτυχαν τα Σχολικά Εργαστήρια, όταν χρησιμοποίησαν ιδιοταγές λογισμικό, χωρίς να διαθέτουν καθόλου χρηματοδότηση από τον προϋπολογισμό, για την υλοποίηση και την υποστήριξη. Αυτός είναι ο λόγος που το νομοσχέδιό μας δεν περιορίζεται στο να καταστήσει υποχρεωτική την χρήση Ελεύθερου Λογισμικού, αλλά προχωρά στην αναγνώριση ότι υπάρχει ανάγκη για την κατάστρωση ενός πραγματοποιήσιμου σχεδίου μετάβασης, στο οποίο το Κράτος θα αναλαμβάνει τα τεχνολογικά θέματα της μετάβασης, κατά έναν οργανωμένο τρόπο, ούτως ώστε να μπορούμε κατόπιν να απολαμβάνουμε τα οφέλη του Ελεύθερου Λογισμικού.

Τελειώνετε με μία ρητορική ερώτηση: 13. "Αν το λογισμικό ανοικτού κώδικα πληρούσε όλες τις προϋποθέσεις που απαιτούνται από τις κρατικές υπηρεσίες, τότε για ποιό λόγο χρειαζόταν να θεσπισθεί ένας νόμος για τη χρήση του; Δεν θα έπρεπε η αγορά να μπορεί να επιλέγει ελεύθερα ποια προϊόντα παρέχουν τα περισσότερα πλεονεκτήματα και τη μεγαλύτερη αξία;"

Συμφωνούμε ότι στον ιδιωτικό τομέα της οικονομίας, πρέπει η ίδια η αγορά να αποφασίζει ποια προϊόντα θα χρησιμοποιήσει, χωρίς να είναι αποδεκτή η Κρατική παρεμβολή σε αυτό το θέμα. Στη περίπτωση, όμως, του Δημόσιου Τομέα, το σκεπτικό δεν είναι το ίδιο: όπως έχουμε ήδη ξεκαθαρίσει, το Κράτος αρχειοθετεί, χειρίζεται και μεταδίδει πληροφορίες που δεν του ανήκουν, αλλά τις οποίες οι πολίτες εμπιστεύθηκαν στο Κράτος, χωρίς να έχουν άλλες εναλλακτικές, σύμφωνα με τις προβλέψεις του Νόμου. Σαν αντίβαρο σε αυτή τη νομική οριοθέτηση, το Κράτος οφείλει να λαμβάνει ακραία μέτρα προφύλαξης για την ακεραιότητα, την εμπιστευτικότητα και την προσβασιμότητα αυτών των πληροφοριών. Η χρήση ιδιοταγούς λογισμικού, γεννά σοβαρές αμφιβολίες κατά πόσον αυτές οι απαιτήσεις μπορούν να καλυφθούν, μη έχοντας στη διάθεσή μας οριστικές αποδείξεις για το θέμα αυτό και, άρα, δεν είναι κατάλληλο προς χρήση από τον δημόσιο τομέα.

Η ανάγκη για την θεσμοθέτηση ενός νόμου βασίζεται, κατ' αρχάς, στη συνειδητοποίηση των

βασικών αρχών, που παρατίθενται πιο πάνω, στον ιδιαίτερο τομέα του λογισμικού. Κατά δεύτερο λόγο, βασίζεται στο γεγονός ότι το Κράτος δεν είναι μία ιδεώδης ομοιογενής οντότητα, αλλά αποτελείται από πολλούς οργανισμούς, με ποικίλους βαθμούς αυτονομίας στη λήψη αποφάσεων. Με δεδομένη, λοιπόν, την ακαταλληλότητα χρήσης του ιδιοταγούς λογισμικού, η απόφαση της θεσμοθέτησης αυτών των κανόνων στο νομικό πλαίσιο, θα εμποδίσει τον κάθε κρατικό υπάλληλο να βασίζεται στη διακριτική του ευχέρεια, θέτοντας σε κίνδυνο τις πληροφορίες που ανήκουν στους πολίτες. Πάνω από όλα, όμως, η ανάγκη υφίσταται, διότι αποτελεί μία ανανεωμένη επαναβεβαίωση για τους τρόπους διαχείρισης και κοινοποίησης των πληροφοριών που χρησιμοποιούμε σήμερα, και βασίζεται στη δημοκρατική αρχή της δημόσιας διαφάνειας.

Σύμφωνα με τις παραπάνω αρχές καθολικής αποδοχής, ο πολίτης έχει το δικαίωμα να γνωρίζει όλες τις πληροφορίες που κατέχει το Κράτος και που δεν καλύπτονται από βάσιμες εξαιρέσεις νόμιμης μυστικότητας. Τώρα, το λογισμικό ασχολείται με τις πληροφορίες και αυτό το ίδιο συνιστά πληροφορίες. Πρόκειται για πληροφορίες σε μία ειδική μορφή, που μπορούν να γίνουν κατανοητές από ένα μηχάνημα, για να εκτελεσθούν διάφορες ενέργειες, αλλά χωρίς να παύουν να αποτελούν πληροφορίες κρίσιμης σημασίας, καθώς ο πολίτης έχει το νόμιμο δικαίωμα να ξέρει, για παράδειγμα, πως γίνεται η υπολογιστική διαχείριση της ψήφου του και πως υπολογίζονται οι φόροι του. Για να επιτευχθεί αυτός ο σκοπός, θα πρέπει να έχει ελεύθερη πρόσβαση στον πηγαίο κώδικα και να είναι σε θέση να αποδείξει ότι τα προγράμματα που χρησιμοποιούνται, για εκλογικούς υπολογισμούς ή για τον υπολογισμό των φόρων του, είναι ικανοποιητικά.

4.5. Διάταγμα περί Μέτρων για την προώθηση της Κοινωνίας της Πληροφορίας στην Ανδαλουσία

Ακολουθούν μερικά από τα άρθρα για το Ελεύθερο Λογισμικό, από το προαναφερθέν Διάταγμα περί μέτρων προώθησης της Κοινωνίας της Πληροφορίας στην Ανδαλουσία [99].

• Άρθρο 11. Εκπαιδευτικό υλικό σε ηλεκτρονική μορφή για υπολογιστές.

1. Όλα τα δημόσια εκπαιδευτικά κέντρα θα διαθέτουν εκπαιδευτικά υλικά και προγράμματα σε ηλεκτρονική μορφή, που θα βασίζονται, κατά προτίμηση, σε Ελεύθερο Λογισμικό. Σε κάθε περίπτωση, τα κέντρα θα παραλαμβάνουν αυτά τα υλικά από την Περιφερειακή Κυβέρνηση της Ανδαλουσίας.

2. Αντιστοίχως, οι διδάσκοντες θα λαμβάνουν κίνητρα για να χρησιμοποιούν αυτά τα ηλεκτρονικά υλικά ή προγράμματα του επίσημου σχολικού προγράμματος, ή για τη χρήση του Διαδικτύου, ειδικά όταν πρόκειται για ανάπτυξη που έγινε με την χρήση Ελεύθερου Λογισμικού.

• Άρθρο 31. Ελεύθερο Λογισμικό

1. Όταν θα γίνεται αγορά υπολογιστών για χρήση σε δημόσια εκπαιδευτικά κέντρα, για διδακτικές δραστηριότητες, θα πρέπει να διασφαλίζεται ότι όλα τα μηχανήματα (*hardware*) είναι συμβατά με λειτουργικά συστήματα που βασίζονται σε Ελεύθερο Λογισμικό. Οι υπολογιστές θα πρέπει να έχουν προεγκατεστημένο όλο το Ελεύθερο Λογισμικό που χρειάζεται για τις συγκεκριμένους σκοπούς για τους οποίους προορίζονται.

2. Ο υπολογιστικός εξοπλισμός που παρέχει η Περιφερειακή Διοίκηση της Ανδαλουσίας για την δημόσια πρόσβαση στο Διαδίκτυο, θα βασίζεται σε προϊόντα Ελεύθερου Λογισμικού.

3. Η Περιφερειακή Διοίκηση της Ανδαλουσίας θα στηρίζει την διάδοση και την προσωπική,

οικιακή και εκπαιδευτική χρήση του Ελεύθερου Λογισμικού. Για τους σκοπούς αυτούς, θα προσφέρεται μία online συμβουλευτική υπηρεσία, για εγκατάσταση και χρήση αυτού του είδους των προϊόντων.

• Άρθρο 49. Σκοπός.

1. Θα προβλέπονται επιδοτήσεις για την ανάπτυξη καινοτόμων projects, που θα διευκολύνουν την ενσωμάτωση του IT και των επικοινωνιών, στην επαγγελματική και εργασιακή εκπαίδευση.

2. Αυτά τα projects θα ακολουθήσουν ένα από τα εξής μοντέλα:

α) Προετοιμασία υλικού και περιεχομένου για επαγγελματική και εργασιακή εκπαίδευση, ώστε να χρησιμοποιηθούν και να διανεμηθούν μέσω του Διαδικτύου, ειδικά για πράγματα που αναπτύχθηκαν με τη χρήση Ελεύθερου Λογισμικού.

β) Πρωτοβουλίες για την εκπαίδευση, που χρησιμοποιούν καινοτόμες μεθόδους, όπως η εξ' αποστάσεως μάθηση και οι μέθοδοι με τις οποίες οι εκπαιδευόμενοι χρειάζεται να παρακολουθούν τα μαθήματα μόνο σε κάποιες περιστάσεις.

5. Παράρτημα Ε'. Creative Commons' Attribution-ShareAlike

Version 3.0 Unported



CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

1. Άδεια

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORISED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Ορισμοί

- a) "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognisably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronisation of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.
- b) "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined below) for the purposes of this License.
- c) "CreativeCommonsCompatibleLicense" means a license that is listed at <http://creativecommons.org/compatiblelicenses> that has been approved by Creative Commons as being essentially equivalent to this License, including, at a minimum, because that license: (i) contains terms that have the same purpose, meaning and effect as the License Elements of this License; and, (ii) explicitly permits the relicensing of adaptations of works made available under that license under this License or a Creative Commons jurisdiction license with the same License Elements as this License.
- d) "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.
- e) "LicenseElements" means the following high-level license attributes as selected by Licensor and indicated in the title of this License: Attribution, ShareAlike.
- f) "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.
- g) "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organisation that transmits the broadcast.
- h) "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a

choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

i) "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

j) "PubliclyPerform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

k) "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. LicenseGrant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a) to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;

b) to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified.";

c) to Distribute and Publicly Perform the Work including as incorporated in Collections; and,

d) to Distribute and Publicly Perform Adaptations.

e) For the avoidance of doubt:

a) Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect

royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;

b) Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,

c) Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a) You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(c), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(c), as requested.

b) You may Distribute or Publicly Perform an Adaptation only under the terms of: (i) this License; (ii) a later version of this License with the same License Elements as this License; (iii) a Creative Commons jurisdiction license (either this or a later license version) that contains the same License Elements as this License (e.g., Attribution-ShareAlike 3.0 US); (iv) a Creative Commons Compatible License. If you license the Adaptation under one of the licenses mentioned in (iv), you must comply with the terms of that license. If you license the Adaptation under the terms of any of the licenses mentioned in (i), (ii) or (iii) (the "Applicable License"), you must comply with the terms of the Applicable License generally and the following provisions: (I) You must include a copy of, or the URI for, the Applicable License with every copy of each Adaptation You Distribute or Publicly Perform; (II) You may not offer or impose any terms on the Adaptation that restrict the terms of the Applicable License or the ability of the recipient of the Adaptation to exercise the rights granted to that recipient under the terms of the Applicable License; (III) You must keep intact all

notices that refer to the Applicable License and to the disclaimer of warranties with every copy of the Work as included in the Adaptation You Distribute or Publicly Perform; (IV) when You Distribute or Publicly Perform the Adaptation, You may not impose any effective technological measures on the Adaptation that restrict the ability of a recipient of the Adaptation from You to exercise the rights granted to that recipient under the terms of the Applicable License. This Section 4(b) applies to the Adaptation as incorporated in a Collection, but this does not require the Collection apart from the Adaptation itself to be made subject to the terms of the Applicable License.

c) If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilising: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv), consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4(c) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

d) Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF

TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability.

EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a) This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b) Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a) Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b) Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

c) If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

d) No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

e) This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work

not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

f) The rights granted under, and the subject matter referenced, in this License were drafted utilising the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

6. Παράρτημα ΣΤ'. Άδεια GNU Free Documentation License

Copyright (C) 2000,2001,2002 Free Software Foundation, Inc. 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject.

(Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them. The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant.

The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible.

You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public. It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of

the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on.
These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

- N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant.

To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties - for example, statements of peer review or that the text has been approved by an organisation as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>. Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.